

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky

Katedra informatiky

Zpracování panoramatických snímků
v Silverlightu

Panorama photos processing in Silverlight

Prohlášení Studenta

„Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.“

V Ostravě 4. května 2011

.....

Jméno Příjmení

Na tomto místě bych chtěl poděkovat vedoucímu práce Ing. Michalovi Radeckému za podnětné a trpělivé vedení, a dále všem, kteří mi během práce naslouchali a přispívali svými nápady a připomínkami.

Abstrakt

Cílem této práce je zmapovat a popsat technologie pro pořizování a zpracování panoramatických snímků. Probrat technologie MS Silverlight a DeepZoom a na základě této technologie navrhnout aplikaci, pro prohlížení panoramatických snímků. Aplikace musí být datově nezávislá, tj. konkrétní obsah musí být jednoduchým způsobem zaměnitelný, jedna aplikace pro více panoramat. Dalším krokem je provést analýzu, návrh a implementaci aplikace. Důraz je kladen na možnost integrace této prohlížečky do dalších aplikací. Součástí práce je i detailní programátorská a uživatelská příručka s popisem použití dané komponenty, pro tvorbu nových panoramat a jejich publikování prostřednictvím této komponenty.

Klíčová slova

Microsoft, DeepZoom, Silverlight, Panorama, Image Composite Editor, DeepZoom Composer, Web, Internet, RIA aplikace

Abstract

Objective of this work is to explore and describe technology for taking and processing panorama photos. Talk over technology MS Silverlight and DeepZoom and on the basis of this technology create application for viewing panorama photos. Application must be data-independent, which means that users can easy change the content of panorama photos in one application. The next step is make analysis, design and implementation of application. Important part is the fact, that application is easy implement to other applications. This work contains also programmer and user guide with description used component for create new panorama photos and their public by this component.

Keywords

Microsoft, DeepZoom, Silverlight, Panorama, Image Composite Editor, DeepZoom Composer, Web, Internet, RIA application

Seznam použitých symbolů a zkratek

Active-X	-	Framework společnosti Microsoft
ASP	-	Active Server Page, skriptovací platforma společnosti Microsoft
DOM	-	Document Object Model
Element	-	Část XML dokumentu
HD	-	High-definition
HTML	-	Hypertext Markup Language
HTTP	-	Hypertext Transfer Protocol
IDE	-	Integrated Development Enviroment, vývojové prostředí
JPEG	-	Joint Photographic Expert Group, formát pro ukládání obrázků
PHP	-	Personal Home Page, skriptovací programovací jazyk
PNG	-	Portable Netwotk Graphics, grafický formát pro bezztrátovou kompresi
QTVR	-	Quick Time Virtual Reality
RIA	-	Rich Internet Application - bohatá Internetová aplikace
TAG	-	Množina značek XML dokumentu
URL	-	Uniform Resource Locator, jednotný lokátor zdrojů
WWW	-	World Wide Web
XHTML	-	Extensible Hypertext Markap Language
XML	-	Extensible Markup Language, jazyk pro strukturovanou reprezentaci dat

Obsah

ÚVOD.....	1
1 VIRTUÁLNÍ PROHLÍDKA.....	2
1.1 ZHOTOVENÍ FOTOGRAFIÍ	2
1.1.1 Fotoaparát.....	2
1.1.2 Stativ.....	3
1.2 VÝBĚR MOTIVU	3
1.2.1 Sférické prohlížení.....	3
1.2.2 Cylindrická projekce	4
1.3 NASTAVENÍ FOTOAPARÁTU	4
1.4 POŘÍZENÍ SNÍMKŮ	4
1.5 ZPRACOVÁNÍ SNÍMKU.....	5
1.6 ZPRACOVÁNÍ PANORAMATICKÉHO SNÍMKU	7
1.6.1 Import panoramatické fotografie	8
1.6.2 Kompozice DeepZoom Snímku.....	9
1.6.3 Export DeepZoom snímku	10
2 TECHNOLOGIE VIRTUÁLNÍ PROHLÍDKY	10
2.1 AKTIVNÍ BODY - HOTSPOTY	11
2.2 NÁSTROJE UMOŽŇUJÍCÍ VIRTUÁLNÍ PROHLÍDKY	11
2.2.1 Adobe Flash Player.....	12
2.2.2 QuickTime player	12
2.2.3 Deval VR	12
2.2.4 JavaScript.....	12
2.2.5 Silverlight	12
3 TECHNOLOGIE .NET.....	13
3.1 ARCHITEKTURA .NET FRAMEWORK.....	13
3.1.1 CLR – Common Language Runtime.....	14
3.1.2 MSIL.....	14
3.1.3 JIT kompilér	14

4	SILVERLIGHT	14
4.1	HLAVNÍ FUNKCE A VÝHODY	15
4.2	ARCHITEKTURA SILVERLIGHT APLIKACE.....	16
4.2.1	XHTML stránka.....	16
4.2.2	XAML soubor	16
4.3	APLIKAČNÍ LOGIKA	17
4.4	HISTORIE VERZÍ	17
4.4.1	Silverlight 1.0	17
4.4.2	Silverlight 2.0	17
4.4.3	Silverlight 3.0	18
4.4.4	Silverlight 4.0	18
4.4.5	Silverlight 5.0	18
4.5	VÝVOJOVÉ PROSTŘEDÍ	19
4.5.1	Visual Studio:	19
4.5.2	Microsoft Expression Blend:	20
5	TECHNOLOGIE DEEPZOOM.....	21
5.1	VYUŽITÍ:	21
5.1.1	Prohlížení fotografií s vysokým rozlišením	21
5.1.2	Panoramatické fotografie.....	21
5.1.3	Reklamní využití	21
5.2	HISTORIE	21
5.3	FUNKCE	21
5.4	DĚLENÍ.....	22
5.4.1	Samostatný obrázek.....	22
5.4.2	Rozptýlený obrázek.....	22
5.4.3	Kolekce Obrázků	23
5.5	VYTVOŘENÍ DEEPZOOM OBRÁZKU.....	23
5.6	NAČTENÍ OBRÁZKU:	23
6	APLIKACE PRO VIRTUÁLNÍ PROHLÍDKY	26
6.1	ANALÝZA.....	26

6.2	NÁVRH.....	27
6.2.1	<i>Posun snímku.....</i>	27
6.2.2	<i>Přiblížení.....</i>	27
6.2.3	<i>Iluze nekonečného obrázku</i>	27
6.2.4	<i>Přechod na jiný snímek</i>	27
6.3	IMPLEMENTACE	28
6.3.1	<i>Přiblížení.....</i>	28
6.3.2	<i>Pohyb.....</i>	28
6.3.3	<i>Iluze nekonečného obrázku</i>	29
6.3.4	<i>Vytvoření hotspotu.....</i>	30
6.3.5	<i>Načítání hotspotu</i>	32
7	NAVRŽENÁ APLIKACE	34
7.1	FUNKCE	35
7.2	VYUŽITÍ	35
7.3	DALŠÍ VÝVOJ	35
	ZÁVĚR	36
	SEZNAM LITERATURY A ZDROJŮ.....	37
	SEZNAM OBRÁZKŮ	40
	OBSAH PŘILOŽENÉHO CD.....	42

Úvod

Pod slovem panorama si každý z nás představí nádherný nezapomenutelný pohled, který poskytuje nevšední zážitek. Od nepaměti je snahou tento pohled zachytit a podělit se o něj s přáteli, kteří neměli to štěstí vidět tuto scenérii na vlastní oči. V minulosti byl jediným nástrojem papír a tužka, pomocí nichž bylo možné panorama částečně zachytit.

Dalším evolučním krokem byl jistě fotoaparát, který dokáže panorama realisticky zachytit, výsledná fotografie je však značně omezena i přes snahy v posledních letech využívat širokoúhlé formáty. Je to dáno tím, že panorama zabírá větší zorné pole, než které můžeme pomocí fotoaparátu zachytit. Samozřejmě můžeme zachytit více snímků, avšak tyto snímky již ztrácí celistvost a výsledný dojem.

Proto následujícím evolučním krokem bylo vytvoření tzv. Virtuální prohlídky, která umožňuje zachytit celou scenérii a následně si ji prohlédnout na jakémkoli počítači s webovým prohlížečem. Výsledný dojem má velmi blízko ke skutečnosti.

1 Virtuální prohlídka

Virtuální prohlídka umožňuje prohlížení panoramatických fotografií, za použití počítače a webového prohlížeče, jež splňuje požadavky dané technologie, která prohlídku zajišťuje. Virtuální prohlídku lze vytvořit několika způsoby, pomocí mnoha nástrojů různých společností. Tato práce je zaměřena na technologie společnosti Microsoft, konkrétně technologie Microsoft Silverlight a DeepZoom.

Cílem této práce je postup, jak virtuální prohlídku vytvořit, od focení fotografií přes jejich úpravu až po finální aplikaci, která prohlídky umožňuje, součástí je i popis zmiňovaných technologií a jejich srovnání s konkurencí. Vše je pečlivě vysvětleno a popsáno s množstvím fotografií, které danou problematiku znázorňují.

1.1 Zhotovení fotografií

Prvním krokem pro vytvoření virtuální prohlídky je zhotovení fotografií, které panorama zachycují. Tento krok je velmi důležitý, protože přesně popisuje, jak mají být fotografie zhotoveny a následně upraveny, tak aby s nimi bylo možno dále pracovat.

Každá technologie umožňující virtuální prohlídky má lehce odlišné požadavky pro zhotovení a zpracování snímků. V této práci je uveden postup zpracování snímku pro technologii Microsoft Silverlight a DeepZoom.

1.1.1 Fotoaparát

Pro zhotovení fotografií je vhodné použít kvalitnější fotoaparát, který umožňuje zhotovit snímky ve vyšším rozlišení, které se pohybuje v řadu desítek Mega-pixelů. V případě, že bude použit fotoaparát s nízkým rozlišením, dojde k výraznému zhoršení kvality fotografie v okamžiku, kdy se uživatel pokusí panorama přiblížit.



Obrázek 1: Použitý fotoaparát

1.1.2 Stativ

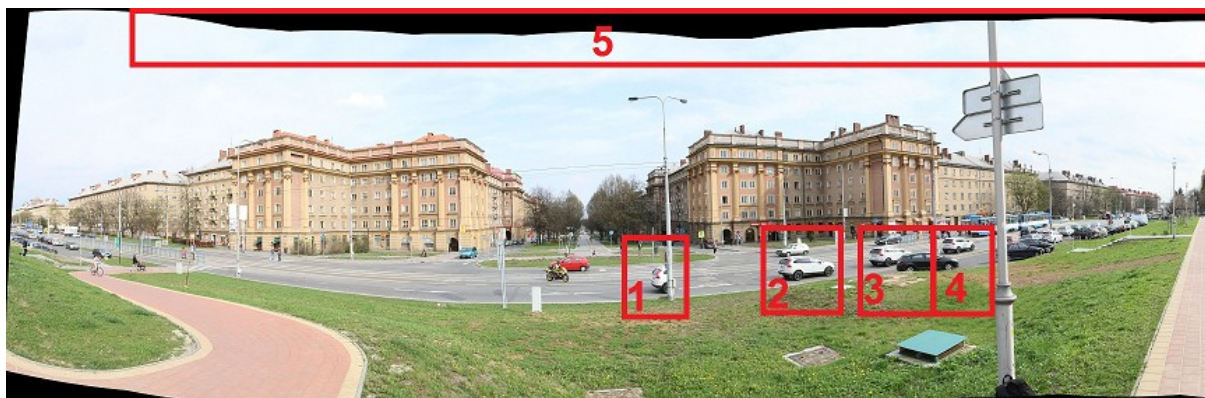
Pro focení panoramat je stativ obrovskou výhodou, jelikož při focení není jednoduché fotoaparát udržet v klidu a v jedné rovině. To je důležité, pokud scéna obsahuje méně světla a fotoaparát má delší expozici. Dále je nutné, aby snímky byly v jedné rovině. Toho lze docílit využitím otáčecího mechanismu stativu. Výsledkem je dokonalejší spojení snímků bez znatelných spojů.

1.2 Výběr motivu

Při výběru motivu je třeba dát pozor na světlo. Motiv by měl být rovnoměrně osvětlen. Mezi fotografováním jednotlivých snímků uběhne velmi dlouhá doba. Proto je dobré vybírat motivy s minimem pohyblivých předmětů, jako jsou lidi, automobily, letadla na obloze nebo samotné slunce, které se může schovat za mraky, či naopak.

V případě, že bude vybrán motiv s velkým počtem pohyblivých předmětů. Je velmi pravděpodobné, že se jednotlivé objekty promítnou do více snímků, tak jak se při focení pohybovaly. Po následném spojení těchto snímků, dojde k tomu, že se na výsledný panoramatický snímek promítnou vícekrát, což se v reálném světě stát nemůže a tento problém kazí celkový dojem virtuální prohlídky.

Tento problém lze vidět na obrázku číslo dvě, kde se projíždějící automobil promítnul do více snímků (bod 1, 2, 3, 4) a po kompozici panoramatu jej lze vidět několikrát. Dalším problémem kterým tento výsledný snímek trpí je fakt, že nebyl použit stativ, což se negativně odrazilo na výsledné kvalitě snímku (bod 5).



Obrázek 2: Ukázka nevhodně použitého motivu.

Virtuální prohlídky lze rozdělit na dva základní typy: Sférické prohlížení a Cylindrickou projekci.

1.2.1 Sférické prohlížení

Je to fotografie, která je spojena z několika samostatných snímků pořízených v určitém sledu tak, aby byl pokryt prostor v celkovém zorném úhlu 360° horizontálně a 180° (někdy označováno jako 360°) vertikálně.

Takto lze prohlížet celý prostor (interiér či exteriér) tak jako by byl uživatel opravdu fyzicky přítomen, bez toho aniž by překročil práh svého domu.

1.2.2 Cylindrická projekce

Je takový typ projekce, který se užívá k poutavým pohledům například na panorama hor a tam, kde není pohled na nebe nebo na zem tak zajímavý. Přesto, že je tento typ projekce složen z více snímků, nemusí nutně zabírat větší zorné pole, než je běžné zorné pole člověka.



Obrázek 3: Cylindrická projekce (obrázek převzat, zdroj č. 24).

1.3 Nastavení fotoaparátu

Z pohledu nastavení fotoaparátu je nejdůležitější, aby všechny snímky měli stejnou úroveň osvětlení a hloubku ostrosti. Je vhodné použít fotoaparát s funkcí panorama. Tato funkce zobrazí na displeji pomocné linky, dále zamkne ostření, expozici a vyvážení bílé. V případě, že fotoaparát tuto funkci nepodporuje, lze obejít focení následovně:

- Zaměřit fotoaparát na nějaký bod, který má osvětlení na průměrné hodnotě.
- Stisknout spoušť do poloviny a počkat. Fotoaparát provede uložení všech parametrů, jako jsou vzdálenost, světlo a podobně.
- Otočit fotoaparát na motiv a stisknout spoušť až na doraz.
- Postup se opakuje pro všechny snímky.

1.4 Pořízení snímků

Jednotlivé snímky by se měly překrývat v asi 20-30% své plochy. Je doporučeno držet se těchto hodnot. Při menším překryvu by mohly nastat problémy se spojením, naopak při větším překryvu dojde k získání velkého množství zbytečných obrazových informací. Je dobré najít na motivu nějaký orientační bod, podle kterého budou jednotlivé snímky na sebe navazovat. Zde je velkou výhodou zmiňovaný stativ a aktivovaná funkce panorama. Snímky budou v jedné rovině a při zpracování programem bude spojení přesnější a nedojde k velkému ořezu výsledného snímku.



Obrázek 4: Ukázka překryvu fotografií (obrázek převzat, zdroj č. 25).

Pro dokonalejší pořízení panoramat je dobré použít panoramatickou horizontální hlavu – jedná se o fotografickou hlavu, určenou k nasazení na stativ a fotografování panoramatických záběrů. Funkčnost panoramatické hlavy spočívá v tom, že dokáže naprosto eliminovat takzvané zkreslení paralaxy, které se u běžných fotografických hlav vyskytuje. Díky tomu je možné fotit dokonalá panoramata třeba i z interiérů, kde je vše velmi blízko a zkreslení paralaxy bude i tak zcela minimální nebo nulové.



Obrázek 5: Panoramatická horizontální hlava (obrázek převzat, zdroj č. 26).

1.5 Zpracování snímku

Po zhotovení jednotlivých snímků s dostatečným překryvem, je dalším krokem spojení těchto snímků v jeden ucelený snímek, jehož šířka je rovna 360° zorného pole. Což znamená, že levý okraj snímku plynule navazuje na pravý okraj snímku.

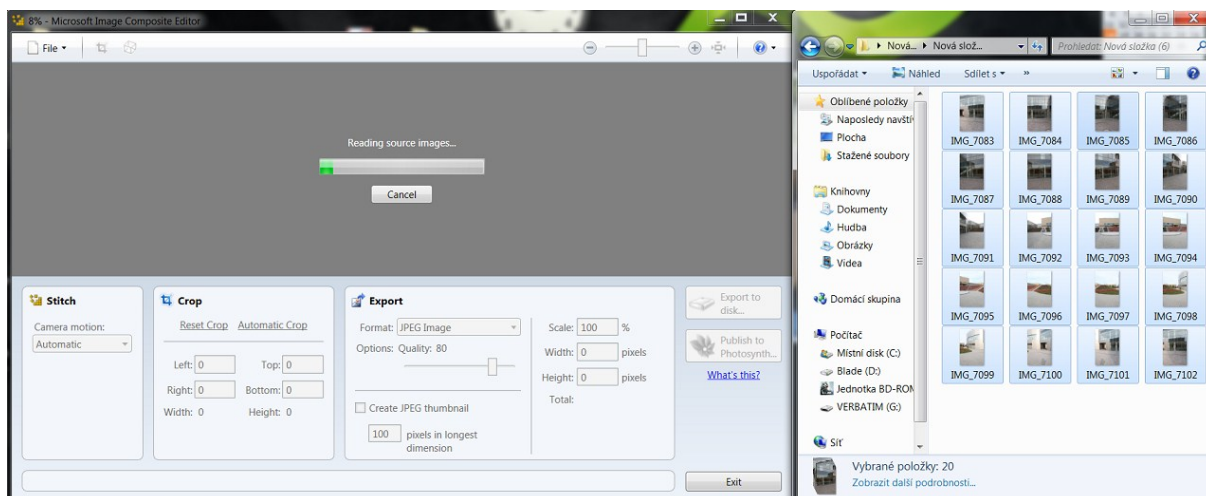
K tomuto účelu slouží celá řada aplikací různých společností. Mezi nejznámější patří například:

- Autopano Pro,
- Hugin
- Microsoft ICE.

Při práci s panoramatickými snímky se nejvíce osvědčil program společnosti Microsoft s názvem Microsoft ICE (Image composite Editor), mezi hlavní výhody patří:

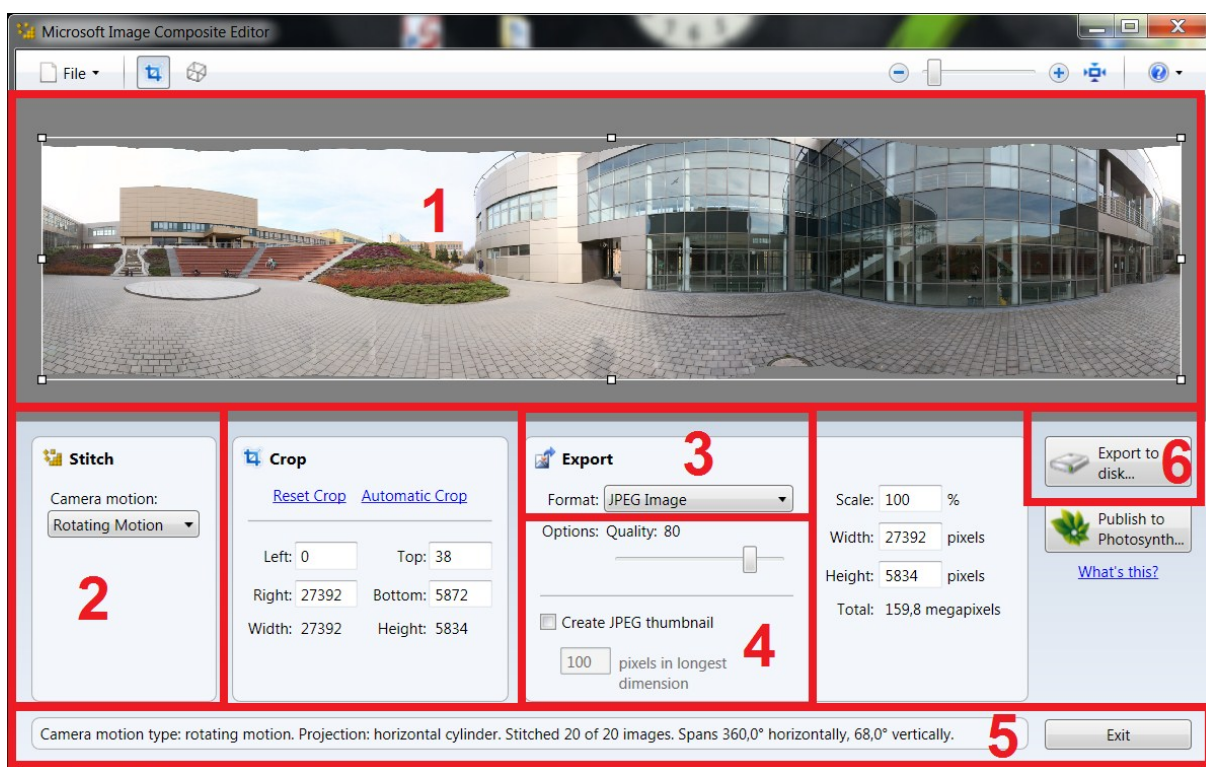
- Akcelerace spojování snímků pomocí více jádrových procesorů.
- Schopnost publikovat, zobrazovat a sdílet panoramata na webových stránkách Photosynth.
- Podpora pro "strukturované panoramata" - panoramata stávající se ze stovek fotografií.
- Vytváření různých typů projekcí: rovinné, sférické, cylindrické.
- Nativní podpora pro 64-bitové operační systémy.
- Široký rozsah výstupních formátů, například JPEG, TIFF, BMP, PNG, HD.

Pro vytvoření požadovaného snímku je nejdříve potřebné přetáhnout snímky z adresáře na pracovní plochu aplikace. Následně dojde k zpracování snímku do panoramatické fotografie.



Obrázek 6: Import snímků do aplikace Microsoft ICE.

Výsledný snímek lze uložit v několika dříve jmenovaných formátech. Aplikace Microsoft ICE disponuje několika zajímavými funkcemi, které napomáhají k optimálnímu spojení snímků.



Obrázek 7: Prostředí aplikace Microsoft ICE.

- 1) Pracovní plocha.
- 2) Výběr vzhledu snímku.
- 3) Výběr výstupního formátu panoramatického snímku.
- 4) Nastavení kvality výstupního snímku.

- 5) *Informace o vygenerovaném snímku, obsahuje informace o typu projekce, počtu snímků, z kterých se panoramatický snímek skládá a dále jaké úhly snímek vyplňuje, ať už se jedná o vertikální nebo horizontální úhel.*
- 6) *Volba pro exportování snímku na disk.*

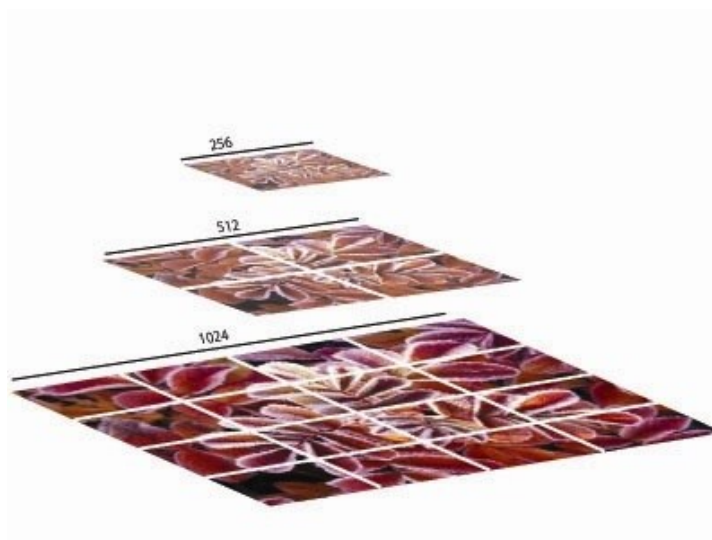
Po zpracování jednotlivých snímků, je výsledek následovný:



Obrázek 8: Výsledný snímek po zpracování v aplikaci Microsoft ICE.

1.6 Zpracování panoramatického snímku

Po vytvoření panoramatického snímku, následuje úprava tohoto snímku pro technologii DeepZoom, která umožňuje virtuální prohlídky. Příprava spočívá ve vytvoření tzv. pyramidy, kde jednotlivá patra představují totožné snímky o různých rozlišeních. To znamená, že vrchol pyramidy má nejnižší rozlišení a analogicky základna pyramidy obsahuje snímek s nejvyšším možným rozlišením. Navíc je každá vrstva rozdělena do jednotlivých dlaždic o stejné velikosti.



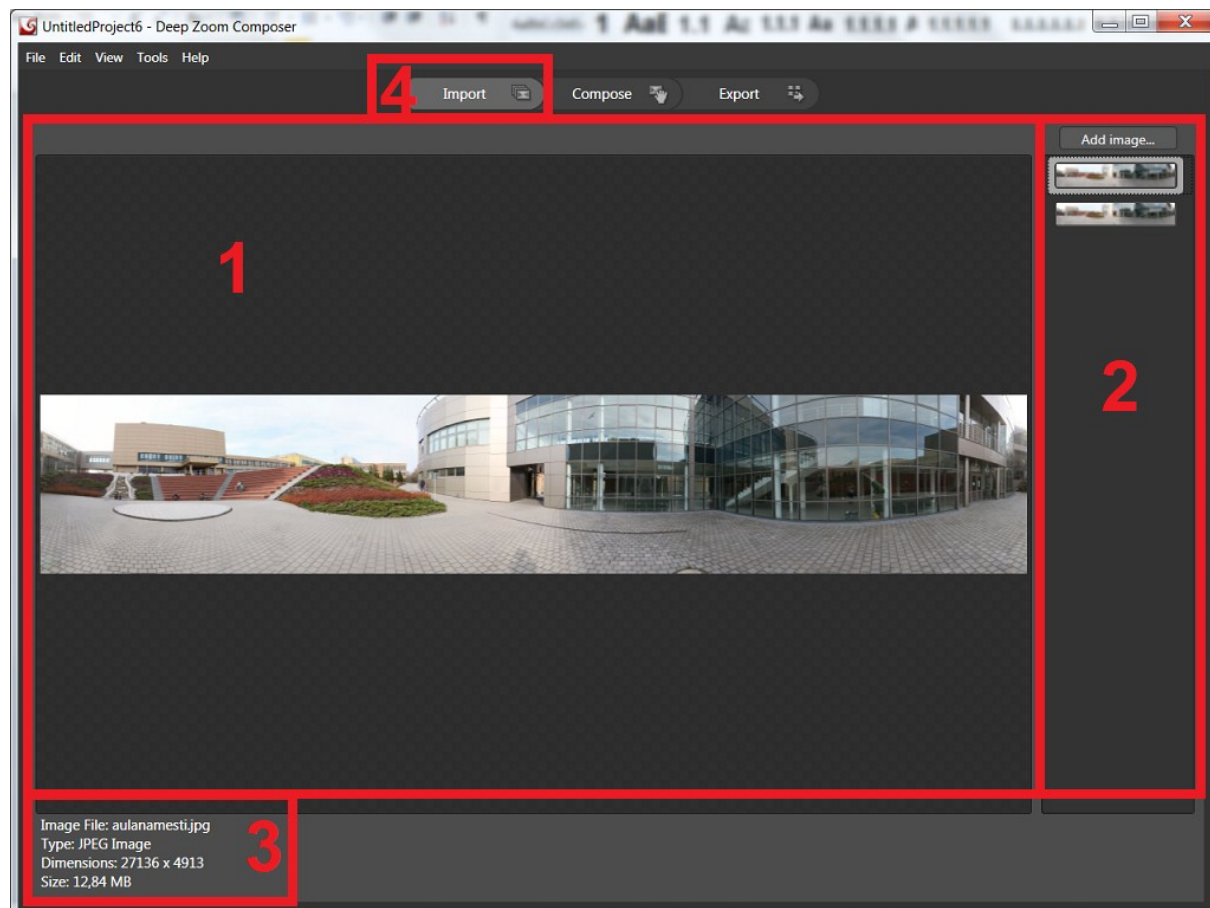
Obrázek 9: Struktura DeepZoom snímku (obrázek převzat, zdroj č. 27).

Jak je vidět tvorba takovéto pyramidy by byla za použití klasických prostředků velmi náročná. Proto Microsoft doporučuje k tvorbě pyramid aplikaci DeepZoom Composer, která tento úkon velmi ulehčuje. Celý proces je složen ze tří částí:

- Import panoramatické fotografie
- Kompozice DeepZoom snímku
- Export DeepZoom snímku

1.6.1 Import panoramatické fotografie

V této části procesu dochází k přidání panoramatického snímku na pracovní plochu. K panoramatickému snímku mohou být přidány i další fotografie nebo obrázek. V takovém případě se již jedná o DeepZoom kolekci, respektive rozptýlený DeepZoom obrázek.

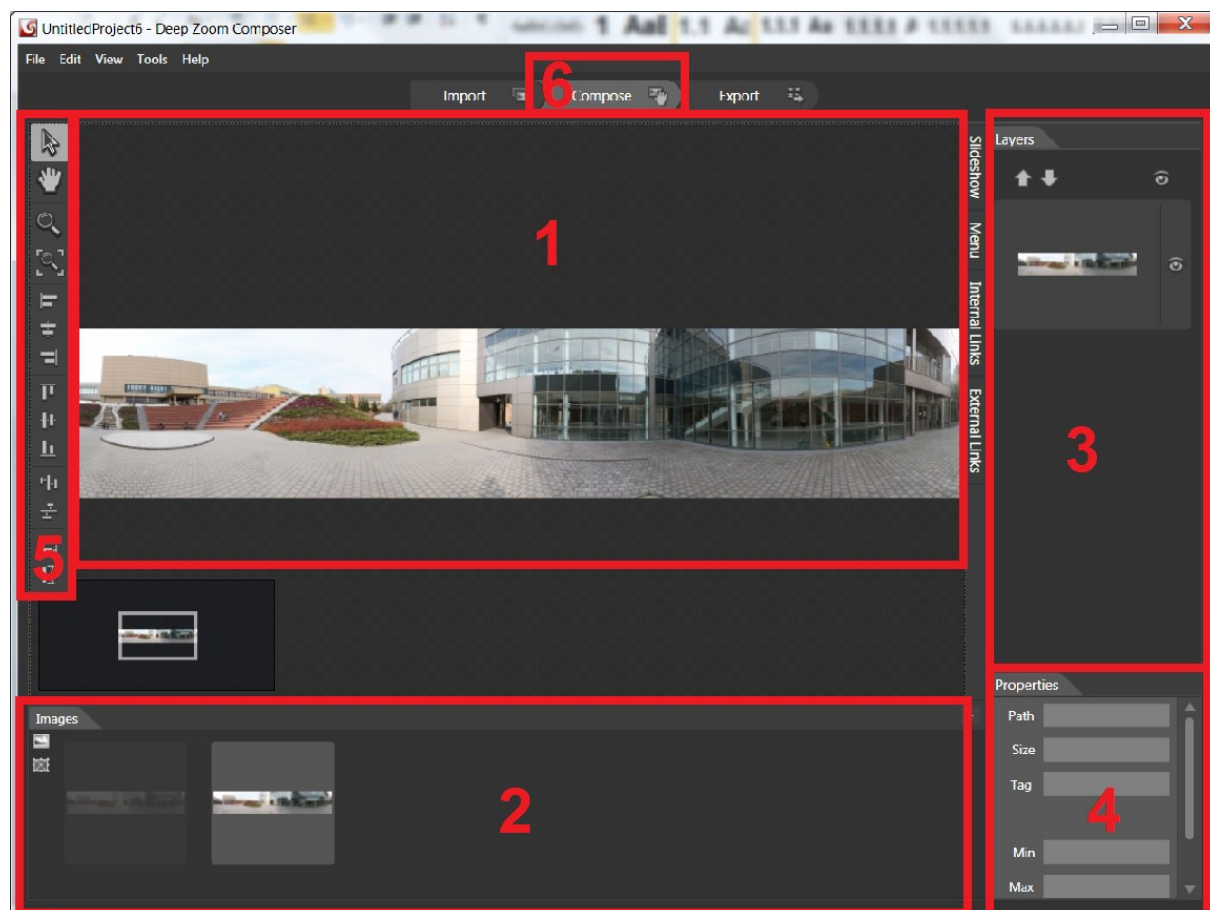


Obrázek 10: DeepZoom Composer – import snímků.

- 1) Plná velikost vybraného snímku.
- 2) Volbou **Add image** přidáváme jednotlivé snímky, ze kterých se bude skládat výsledný DeepZoom obrázek. Můžeme přidávat jakékoliv snímky o jakémkoli rozlišení.
- 3) Informace o vybraném snímku, Název, typ rozlišení a velikost.
- 4) Aktuální Volba - Import snímků.

1.6.2 Kompozice DeepZoom Snímku

Po přidání snímků následuje jejich kompozice. V této části jsou přidány na pracovní plochu snímky ze spodní nabídky s názvem Images. Jednotlivé snímky jsou uloženy v samostatných vrstvách, tak jako je tomu u vektorových grafických nástrojů, z čehož plyne, že se snímky mohou libovolně překrývat. Následně je určen výsledný vzhled DeepZoom snímku.

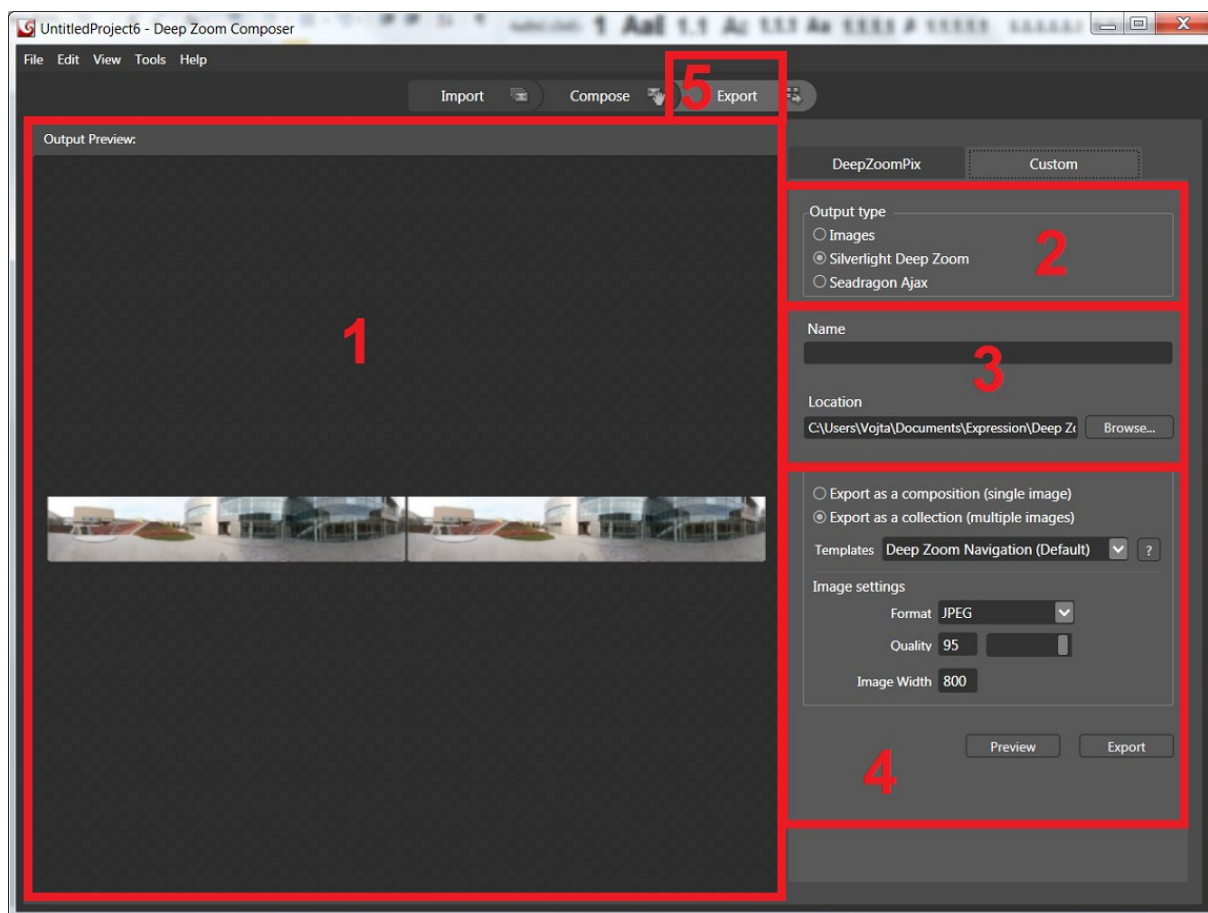


Obrázek 11: DeepZoom Composer – kompozice snímku.

- 1) Pracovní plocha.
- 2) Nabídka dostupných snímků
- 3) Přepínač jednotlivých vrstev, každý snímek se nachází v samostatné vrstvě.
- 4) Vlastnosti DeepZoom obrázku.
- 5) Nabídka funkcí, které umožňují úpravy snímku jako je, přiblížení, oddálení, vyplnění plochy apod.
- 6) Aktivní záložka kompozice snímku.

1.6.3 Export DeepZoom snímku

V poslední části s názvem export probíhá nastavení výstupního formátu snímku. Nachází se zde také volba pro nastavení, zda má být DeepZoom snímek exportován jako jeden snímek nebo jako kolekce s kterou může uživatel v Silverlightu dále pracovat. To znamená, že s jednotlivými snímky může být později dále manipulováno. V případě, že snímek bude exportován jako, samostatný snímek nebude pozdější manipulace s jednotlivými snímky kolekce možná.



Obrázek 12: DeepZoom Composer – Export snímku.

- 1) *Náhled výsledného DeepZoom obrázku.*
- 2) *Výběr výstupního formátu.*
- 3) *Nastavení jména a umístění výsledného snímku.*
- 4) *Poslední volbou je možnost **preview**, jež umožňuje prohlédnout si DeepZoom obrázek v prohlížeči a volba **export**, která exportuje obrázek na příslušné místo.*

2 Technologie virtuální prohlídky

Po vytvoření panoramatického obrázku následuje vytvoření aplikace, která umožňuje Virtuální prohlídku. Virtuální prohlídka je speciální typ sférické fotografie, která pokrývá prostor v celkovém zorném úhlu 360° horizontálně a pokud možno 180° vertikálně. Pořízení snímků probíhá z jednoho místa otáčením fotoaparátu kolem vlastní osy.

Takto nasnímaný prostor, si může uživatel následně prohlížet, tak jako by byl opravdu fyzicky přítomen. Uživatel může sám interaktivně pomocí myši zvolit, kterým směrem a pod jakým úhlem

pohledu chce prostory sledovat. Vidí tak vše kolem sebe, nad i pod sebou, přiblížením si může prohlédnout zajímavé detaily a naopak oddálením získá širokoúhlý pohled v širším kontextu. Na obrázku číslo 13. je názorně vidět ukázka aplikace umožňující virtuální prohlídku.



Obrázek 13: Virtuální prohlídka (obrázek převzat, zdroj č. 28).

Barevně je vyznačen výřez, ve kterém si lze prohlížet celkový prostor (černobíle). Přehrávač vždy zobrazuje pouze určitý úhel pohledu, který lze při prohlížení měnit. Barevná část obrázku je fakticky to, co uživatel virtuální prohlídky na svém monitoru vidí.

2.1 Aktivní body - Hotspoty

Jednotlivé virtuální prohlídky lze mezi sebou vzájemně propojit aktivními body - tzv. Hotspoty. Kliknutím na tyto body, které bývají umístěny v místech, kde lze očekávat další virtuální prohlídku, např. v oblasti dveří, koncích ulic apod., je možno přecházet mezi jednotlivými virtuálními prohlídkami z jedné scény do druhé.

Další využití hotspotu je poskytnutí informací o objektu, jenž se nachází přímo ve virtuální prohlídce. Například virtuální prohlídka galerie obsahuje různá umělecká díla, a ke každému uměleckému dílu lze přidat hotspot. Při kliknutí na tento hotspot dojde k otevření okna prohlížeče a zobrazení informací o konkrétním uměleckém díle.

2.2 Nástroje umožňující virtuální prohlídky

K dispozici je celá řada různých nástrojů, které umožňují virtuální prohlídky snadno prohlédnout. Každá technologie vyžaduje specifický přístup, jako je rozdílné vývojové prostředí nebo programovací jazyk, který daná technologie využívá. Navíc jsou zmíněné nástroje vzájemně nekompatibilní, což znamená, že postup, jež je uveden v této práci nemusí platit pro každou níže uvedenou technologii.

2.2.1 Adobe Flash Player

Je software určený pro prohlížení animací a filmů pomocí webového prohlížeče. Flash player je široce distribuovaný proprietární multimediální a aplikační přehrávač, původně vytvořený společností Macromedia, po jeho získání v současnosti vyvíjený a distribuovaný společností Adobe. Používá soubory typu SWF, které mohou být vytvořeny nástrojem Adobe Flash autothoring tool, Adobe Flex a další nástroje společnosti Macromedia. Flash Player podporuje skriptovací jazyk nazvaný ActionScript, který je založen na ECMAScript.

Flash Player je dostupný jako plugin pro vícere verze webových prohlížečů (např. Mozilla, Firefox, Opera, Safari), nebo jako Active-X pro Internet Explorer na různých platformách. Adobe uvádí, že každá verze pluginu je zpětně kompatibilní, pouze s výjimkou bezpečnostních změn zavedených v rámci verze 10.

2.2.2 QuickTime player

Je multimediální prostředí vyvinuté společností Apple Inc. schopné práce s mnoha formáty digitálního obrazu, klipů, zvuku, textu, animace, hudby a několika typy interaktivních panoramatických obrazů (QuickTime VR).

2.2.3 Deval VR

Deval VR je 3D víceúčelový prohlížeč pro Windows, který umožňuje přehrávat interaktivní obsah. Pomocí tohoto nástroje je možné zobrazit všechny hlavní typy panoramatických snímků: kubických, sférické a cylindrické panorama. Navíc je plně podporován formát QTVR. K dispozici jsou dva programy pro zobrazení panoramatických snímků: Deval VR plugin, pro zobrazení obsahu v internetovém prohlížeči a Deval VR player, který umožňuje zobrazit obsah z lokálních zdrojů, jako je pevný disk nebo CD. Obě verze aplikace jsou k použití zdarma.

2.2.4 JavaScript

JavaScript je multiplatformní, objektově orientovaný skriptovací jazyk, jehož autorem je Brendan Eich z tehdejší společnosti Netscape. JavaScript byl původně obchodní název implementace společnosti Netscape, kde byl vyvíjen nejprve pod názvem Mocha, později LiveScript, ohlášen byl společně se společností Sun Microsystems v prosinci 1995 jako doplněk k jazykům HTML a Java. Pro verzi firmy Microsoft je použit název JScript. JScript je podporován platformou .NET.

Program v JavaScriptu se obvykle spouští až po stažení WWW stránky z Internetu (tzv. na straně klienta), na rozdíl od ostatních jiných interpretovaných programovacích jazyků (např. PHP a ASP), které se spouštějí na straně serveru ještě před stažením z Internetu. Z toho plynou jistá bezpečnostní omezení, JavaScript např. nemůže pracovat se soubory, aby tím neohrozil soukromí uživatele.

2.2.5 Silverlight

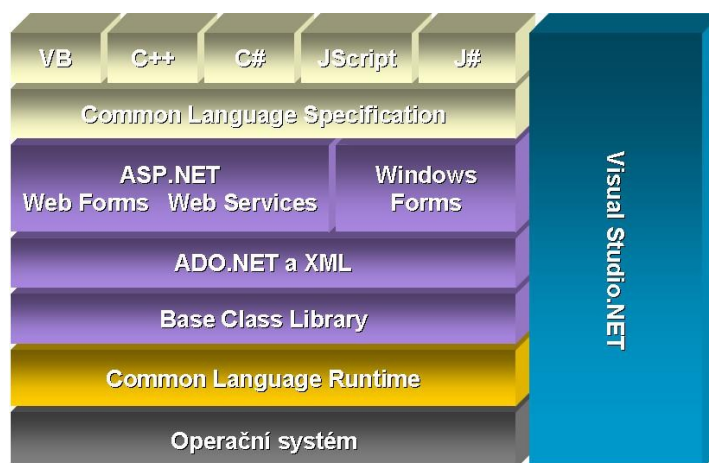
Je plugin vytvořený společností Microsoft pro prohlížeče Windows Internet Explorer, Firefox, Opera, Chrome a Safari pro vytváření interaktivních webových aplikací. Nejvýznamnější skutečností je, že silverlight je založen na zredukované verzi runtime jazyků (common language runtime, CLR) .NET. Tato technologie je použita i v ukázkové aplikaci, která umožňuje prohlížení virtuálních prohlídek.

3 Technologie .NET

V dnešní době jsou vyvíjeny složité aplikace, které svým rozsahem převyšují schopnosti jednoho člověka. Na vývoji aplikací se většinou podílí více programátorů. Při své práci využívají (znovu používají) části (komponenty) svých aplikací nebo aplikace jiných firem. Programátor tedy již nemůže spoléhat jen sám na sebe, tak jak to bylo v dobách operačního systému MS-DOS.

.NET Framework funguje doslova jako substrát, na kterém lze pěstovat software. Jeho jádro je založené na principech objektově orientovaného programování a všechny základní služby zpřístupňuje široké škále programovacím jazykům. .NET Framework automaticky podporuje třídy, metody, vlastnosti, konstruktory, události, polymorfismus atd. Ve výsledném efektu to znamená, že není podstatné, ve kterém programovacím jazyce komponenty vytváříme případně, jaké komponenty používáme. .NET Framework také řeší některé problémy související s bezpečností.

3.1 Architektura .NET Framework

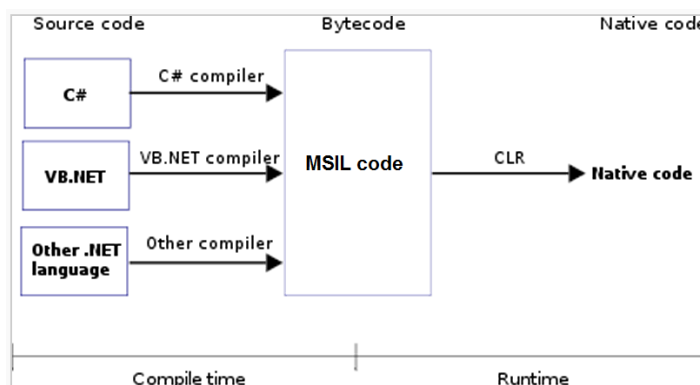


Obrázek 14: Architektura .NET (obrázek převzat, zdroj č. 29).

Na nejnižší úrovni se nachází CLR- Common Language Runtime realizující základní infrastrukturu, nad kterou je framework vybudován. Nad CLR se nachází několik hierarchicky umístěných knihoven. Ty jsou rozděleny do jmenných prostorů. Základem je knihovna nazvaná Base Class Library. Nad ní je knihovna pro přístup k datům a práci s XML soubory. Poslední vrstvou je sada knihoven usnadňující práci s uživatelským rozhraním. Je rozdělena do dvou skupin: pro usnadnění vytváření webových aplikací a pro vytváření klasických aplikací. Poslední vrstvu tvoří nelimitovaná množina programovacích jazyků. Jejich základní vlastnosti definuje CLS – Common Language Specification. V současné době jsou firmou Microsoft podporovány čtyři jazyky Visual Basic, C++, C# a Jscript. Tato množina, ale není uzavřena a jakýkoliv výrobce ji může rozšířit.

3.1.1 CLR – Common Language Runtime

Common Language Runtime je speciální run time prostředí, které poskytuje základní infrastrukturu pro Microsoft .NET Framework. Toto prostředí je místo, kde je zdrojový kód kompilován do přechodného jazyka MSIL. V okamžiku, kdy je program spuštěn, kód MSIL je přeložen do nativního jazyka operačního systému pomocí JIT kompilátoru.



Obrázek 15: Překlad kódu do nativního kódu operačního systému (obrázek převzat, zdroj č. 30).

3.1.2 MSIL

MSIL (Microsoft Intermediate Language) je procesorově nezávislý jazyk podobný assembleru. Oproti assembleru je však mnohem vyspělejší. Umí pracovat s objekty, volat virtuální metody, pracovat s prvky pole nebo zpracovávat výjimky. Důvodem pro zavedení tohoto jazyka je snaha o jednoduché přenášení existujícího kódu mezi různými platformami. Hlavní výhodou použití intermediárního jazyka je platformní nezávislost.

3.1.3 JIT kompilér

JIT je zkratka anglického označení Just In Time, který označuje speciální metodu překladu využívající různé techniky pro urychlení běhu interpretovaných programů. Program, který je spuštěn a prováděn, může být interpretem v době provádění přeložen přímo do nativního strojového kódu počítače, na kterém je prováděn, čímž dochází k urychlení jeho provádění. Negativem této techniky je prodleva, kterou interpret stráví překladem do nativního kódu. Výhodou je, že je kód možné lépe optimalizovat pro daný procesor a využít jeho rozšířených instrukcí.

4 Silverlight

Microsoft Silverlight je moderní technologie pro tvorbu dynamického obsahu webových stránek, která umožňuje v rámci kontextu kombinovat klasické textové prvky, vektorovou a rastrovou grafiku, animace a také i video.

Silverlight je poměrně nová technologie, která byla veřejnosti představena v dubnu roku 2007. V době svého vypuštění ovšem neměla lehkou pozici. Na internetovém trhu totiž byla neohrožitelnou jedničkou konkurenční technologie Adobe Flash. I přesto se ovšem dokázal Silverlight prosadit a dnes se nabízí již ve své páté verzi. Dostupný je jako plugin pro řadu prohlížečů (Internet Explorer, Firefox, Safari, Opera, Chrome) i operačních systémů (Windows, Mac OS X, Linux pod názvem Moonlight).

Silverlight je přímý konkurent technologie Adobe Flash (dále pouze Flash). Podobně jako Flash umožňuje i Silverlight vytvářet interaktivní obsah, který běží u klienta, a podporuje dynamickou grafiku, média a animace, které daleko přesahují možnosti obyčejného HTML. Silverlight se může oproti technologii Flash pochlubit několika architektonickými funkcemi, které Flash nemá - nejvýznamnější je skutečnost, že je založený na zredukované verzi společného runtime jazyků (common language runtime, CLR) .NET, a umožňuje proto vývojářům psát kód u klienta v čistém C#.

Aby Silverlight dokázal překročit meze standardních webových stránek, používá k tomu odlehčený zásuvný modul (lightweight plugin) prohlížeče.

Předností modelu plugin je to, že uživateli stačí nainstalovat jen jedinou komponentu. Jakmile je plugin nainstalovaný, bude prohlížeč moci zpracovávat nepřerušovaně jakýkoli obsah, který tento plugin používá.

Silverlight je navržený tak, aby dokázal překonat limity HTML a umožnil vývojářům vytvářet graficky lepší a interaktivní aplikace. Silverlight ovšem není pro vývojáře vhodná cesta, jak překonat bezpečnostní "sandbox" prohlížeče, což je izolovaný prostor, který slouží k odstínění aplikace od přímého přístupu k systému. Je tak zajištěna ochrana proti škodlivým aplikacím. (isolated storage). Izolované úložiště funguje podobně jako cookies v obyčejné webové stránce. Soubory jsou oddělené podle webů a aktuálního uživatele, jejich velikost je značně omezena.

4.1 Hlavní funkce a výhody

- Velmi široká podpora prohlížečů.
- Odlehčený zásuvný modul - Velikost instalačního souboru je přibližně 4 MB.
- Dvourozměrné kreslení - Silverlight poskytuje bohatý model pro dvourozměrné kreslení. Dále poskytuje možnost odpovídat na události například kliknutí myši na část grafiky. Což výrazně zvyšuje interaktivnost celé aplikace.
- Animace - Silverlight má animační model založený na čase, který umožňuje definovat, co se má stát, a jak dlouho to má trvat. Plugin silverlightu zvládá i details, jako jsou interpolace mezilehlých hodnot nebo výpočet rychlosti přehrávání snímků (frame-rate).
- Média - Silverlight umožňuje přehrávání Windows Media Audio (WMA), Windows Media Video (WMV7–9), MP3 audio a VC-1 (podporuje high-definition).
- CLR - Silverlight zahrnuje zredukovanou verzi CLR, garbage collector, kompilátor JIT (just-in-time) a podporou generiky. V mnoha případech mohou vývojáři využít kód původně napsaný pro úplný CLR .NET a po mírných změnách jej znovu využít v silverlight aplikaci.
- Interakce s webovými službami - Aplikace Silverlight mohou volat webové služby ASP.NET, nebo webové služby WCF (Windows Communication Foundation). Mohou také přes HTTP odesílat ručně vytvořené požadavky XML.

4.2 Architektura Silverlight aplikace

Silverlight aplikace se skládá ze tří částí:

- XHTML stránka
- XAML soubor
- soubory s aplikační logikou

4.2.1 XHTML stránka

XHTML stránka je hostitelem Silverlight aplikace. Výsledná XHTML může vypadat následovně



Obrázek 16: XHTML stránka

Na tomto příkladu lze vidět, že je aplikace reprezentovaná tagem form. Což znamená že, aplikace může být implementována v jakékoliv již existující XHTML stránce.

4.2.2 XAML soubor

XAML je značkovací jazyk pro definici uživatelského rozhraní. Jedná se v podstatě o XML soubor. V XAML souboru se nadefinuje uživatelské rozhraní a handlers událostí. Níže je uveden výšek z kódu XAML z ukázkové aplikace:

XAML kód:

```
<UserControl
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006" mc:Ignorable="d"
    x:Class="SilverlightApplication4.MainPage"

    d:DesignWidth="1307" d:DesignHeight="873" xmlns:dataInput="clr-
namespace:System.Windows.Controls;assembly=System.Windows.Controls.Data.Input"
    MouseLeftButtonUp="UserControl_MouseLeftButtonUp">

    <Grid Name="backGrid" MouseLeftButtonUp="backGrid_MouseLeftButtonUp">
        <Grid.Background>
            <LinearGradientBrush EndPoint="0.5,1"
MappingMode="RelativeToBoundingBox" StartPoint="0.5,0">
                <GradientStop Color="White"/>
                <GradientStop Color="Black" Offset="1"/>
                <GradientStop Color="White" Offset="0.22"/>
            </LinearGradientBrush>
        </Grid.Background>
    </Grid>
</UserControl>
```

4.3 Aplikační logika

Pokud je řeč o Silverlight 1.0 tak se aplikační logika omezuje na javascriptové funkce, které jsou ze Silverlight aplikace volány. Tyto funkce mohou být umístěny v externích souborech nebo přímo v XHTML stránce.

V Silverlight 1.1 je situace úplně odlišná. Přímě v XAML souboru je možno definovat **code behind** zdroj. Tím zdrojem může být buď kompilovaný kód nebo skript. Z kompilovaných jazyků je podporován C# a VB.NET. Jejich běh zajišťuje CLR integrovaný v pluginu.

4.4 Historie verzí

I přesto, že je technologie velmi mladá, dočkala se za čtyři roky své existence již páté verze. Zde jsou uvedeny nejdůležitější prvky, které jednotlivé verze přinesly.

4.4.1 Silverlight 1.0

První verze technologie Silverlight byla představena v roce 2007. Aplikace byly spouštěny vyvoláním ovládacího prvku Silverlight, který následně načetl soubor XAML. Pro obsluhu událostí deklarovaných prvků a definici aplikační logiky, byl zvolen jazyk JavaScript. Společně s první verzí byla zároveň představena také alfa verze 1.1, která byla ve finální fázi přechíslována na Silverlight 2.0.

4.4.2 Silverlight 2.0

Druhá verze přinesla podstatné změny v oblasti aplikační logiky. JavaScript na pozadí byl doplněn jazyky platformy .NET, kterými jsou například C# nebo Visual Basic. Použitím těchto jazyků tak bylo docíleno přístupu k takřka všem knihovnám .NET Frameworku a byla tak rozšířena funkcionální technologie.

Společně s novými jazyky se změnil i způsob sestavení aplikace samotné. Deklarativní soubor XAML je společně s kódem na pozadí zkompilován do .NET assembly, která je zkomprimována pomocí ZIPu a uložena do souboru XAP. Avšak i přesto lze z řízeného kódu Silverlightu přistupovat k celé struktuře stránky a určitým prvkům jazyka HTML (tedy k celému DOM – Document Object Model) díky speciálním wrapperům, které komunikaci zajišťují.

Díky použitým knihovnám přibyla rovněž podpora podmnožiny technologie WPF, která obsahuje nástroje pro kreslení, podporu médií a objekty WPF pro animaci. Dále přibyla podpora vizuálních prvků (TextBox, CheckBox, Slider...), manažerů rozložení (Grid, StackPanel...) a také prvky pro práci s daty (DataGrid, ListBox...). Navíc jsou všechny tyto prvky skinovatelné díky použitým šablonám.

4.4.3 Silverlight 3.0

Verze 3 byla představena v červenci roku 2009. Tato verze se stala zlomovou ve smyslu razantního zmenšení rozdílu mezi webovou a desktopovou aplikací. Od této doby již mohou Silverlight aplikace běžet i mimo prohlížeč v takzvaném režimu Out of Browser (OOB). Uživatel si může aplikaci nainstalovat do lokálního operačního systému, vytvořit si zástupce na pracovní ploše a v podstatě nepozná rozdíl od klasické aplikace. Ve třetí verzi se objevují i prvky pro snazší provádění animací, resp. pro jejich nelineární běh. Nyní lze pomocí tzv. Animation Easing například simulovat odražející se předmět na úrovni XAMLu bez nutnosti programového zásahu do kódu na pozadí.

4.4.4 Silverlight 4.0

Verze 4 byla oznámena dne 18. listopadu 2009. Verze 4 nabízí využití webových kamer a mikrofonů, přináší podporu pro off-line přehrávání chráněného obsahu a přibyla funkce renderování HTML obsahu v oknech aplikací. Novinkou je i 60 nových ovládacích prvků, plná podpora tisku, zlepšená funkčnost bindingu, vylepšené formátování textu, možnosti přidělení vyššího oprávnění a podpora pro snazší tvorbu RIA aplikací.

4.4.5 Silverlight 5.0

Verze 5 byla oznámena dne 2. prosince 2010. První beta vydání 5 verze by mělo být v první polovině roku 2011, a v druhé polovině je naplánováno vydání finální verze. Mezi nejzajímavější novinky ve verzi 5 patří podpora 64bitových systémů, nové funkce pro přehrávání multimediálního obsahu, využití procesoru grafické karty (GPU) pro realizaci 2D a 3D obsahu (hardwarová akcelerace H.264), vylepšená podpora pro DRM, lepší podporu dálkových ovladačů při přehrávání médií a další.

4.5 Vývojové prostředí

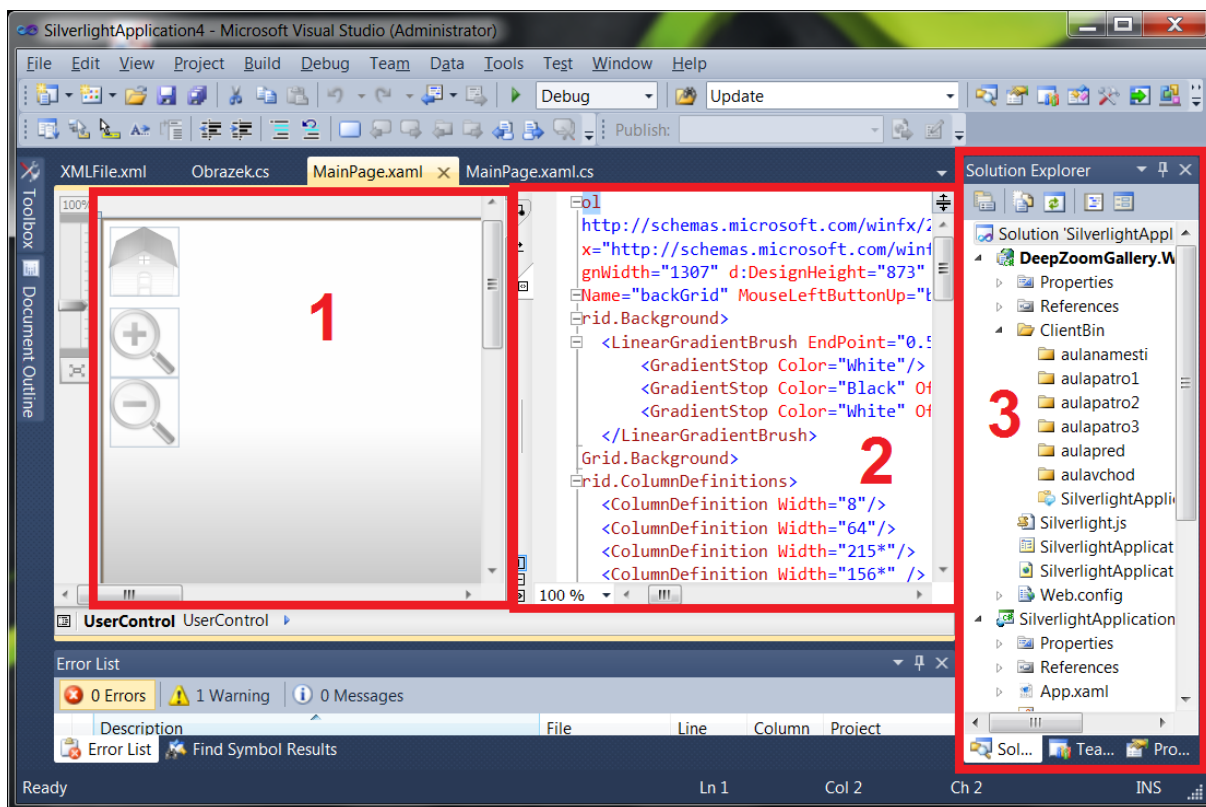
Vývojové prostředí (zkratka IDE, Integrated Development Environment) je software usnadňující práci programátorů. Obsahuje editor zdrojového kódu, kompilátor, případně interpret a také debugger. Pokud se jedná o nástroj pro objektově orientované programování, může obsahovat také object browser.

Vývojová prostředí jsou navržena tak, aby zvýšila produktivitu programátora pomocí provázaných komponent s podobným uživatelským rozhraním. To znamená, že programátor nemusí tolikrát přepínat mezi jednotlivými módy jako při použití individuálních vývojových nástrojů. Nicméně, protože IDE je ze své podstaty komplikovaným softwarem, vysoká produktivita se projeví až po získání určitých zkušeností.

4.5.1 Visual Studio:

Je vývojové prostředí (IDE) od společnosti Microsoft. Může být použito pro vývoj konzolových aplikací a aplikací s grafickým rozhraním spolu s aplikacemi Windows Forms, webovými stránkami a webovými aplikacemi jak ve strojovém kódu, tak v řízeném kódu na platformách Microsoft.

Jedná se o základní vývojový nástroj, aktuálně ve verzi 2010, který slouží k definici chování celé aplikace.



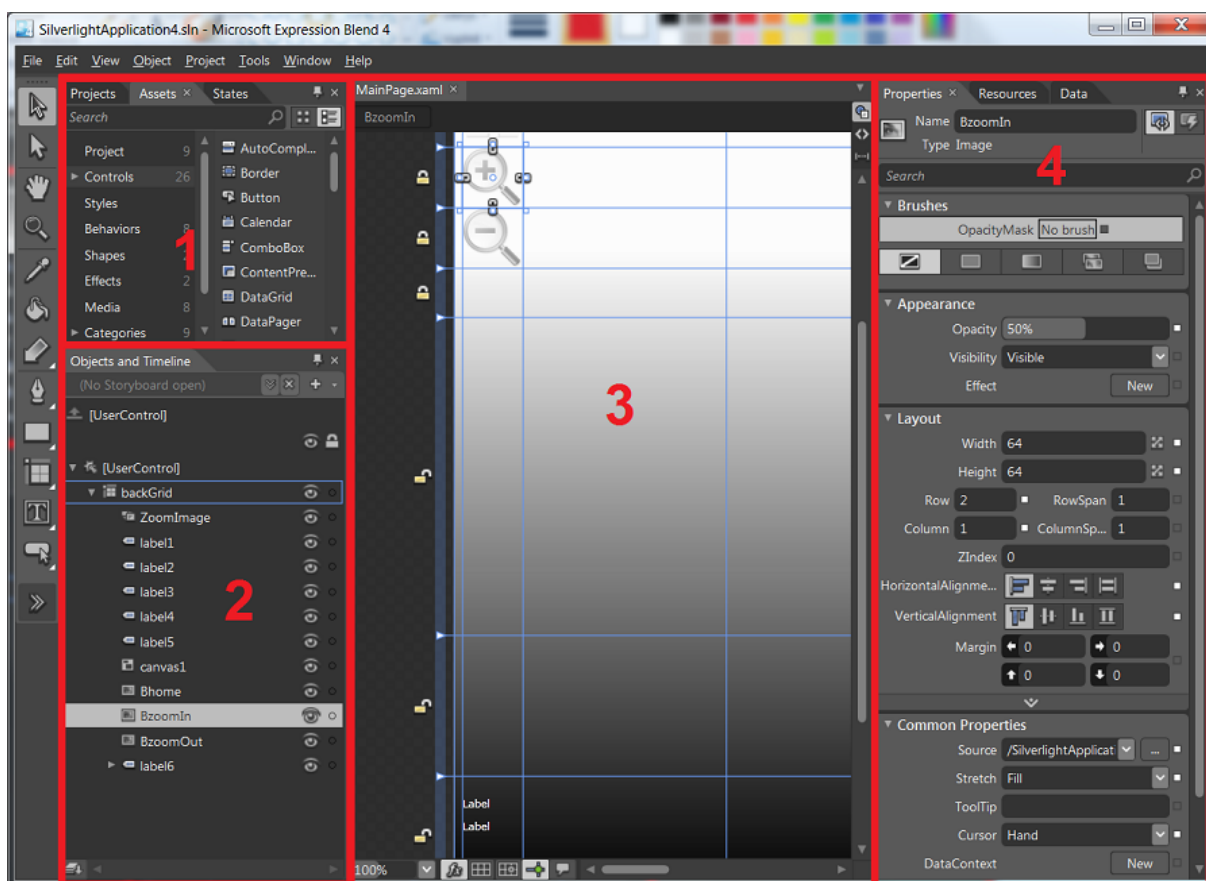
Obrázek 17: Prostředí Visual Studio

- 1) *Náhled vzhledu aplikace.*
- 2) *Kód XAML, který definuje vzhled aplikace.*
- 3) *Solution explorer umožňuje přístup k celé struktuře projektů v rámci týmového vývoje.*

4.5.2 Microsoft Expression Blend:

Microsoft Expression Blend je nástroj pro návrh uživatelského rozhraní vyvinutý a prodáváný společností Microsoft pro vytváření grafických rozhraní pro webové a desktopové aplikace, které kombinují vlastnosti těchto dvou typů aplikací. Návrh aplikace je prováděn v jazyce XAML.

Vývojové prostředí je určeno především designerům k tvorbě uživatelského rozhraní, animací a dalších prvků nesouvisejících s logikou aplikace.



Obrázek 18: Prostředí Microsoft Expression Blend.

- 1) *ToolBox – zde se nachází komponenty jako tlačítko kalendář apod.*
- 2) *Panel obsahující objektový model aktuálního projektu. Můžeme zde vidět celou hierarchii prvků a způsob, jakým jsou jednotlivé prvky do sebe navzájem poskládány.*
- 3) *Pracovní plocha.*
- 4) *Property jednotlivých komponent, zde můžeme nastavit jejich vlastnosti.*

Cílem tedy je, že výsledkem práce návrháře v nástroji Blend je tvář aplikace (soubor XAML), které vývojář vdechne život pomocí nástroje Visual Studio a kódu na pozadí (soubory CS, VB apod.). Oba tak pracují na své izolované části, používají nástroje jim bližší a dohromady tvoří společnou aplikaci nad jednotnou platformou.

Jelikož nejsou hranice mezi aplikační logikou a vzhledem striktně dány, lze v omezené míře použít oba dva tyto nástroje i k úkolům toho druhého z dvojice.

5 Technologie DeepZoom

DeepZoom poskytuje možnost, jak interaktivně zobrazovat snímky s vysokým rozlišením v řádu několika Giga-pixelů. Umožňuje přibližovat a oddalovat snímek velmi rychle, bez vlivu na výkon aplikace. DeepZoom poskytuje plynulé načítání a posouvání, jež je docíleno použitím speciálně upraveného snímku, který je rozdělen do několika vrstev s příslušným rozlišením. Pohyb mezi těmito vrstvami je doprovázen pružnou animací, která zajišťuje plynulost přechodu.

5.1 Využití:

DeepZoom lze využít k různým účelům, od jednoduchých prohlížeček fotografií, přes virtuální prohlídky, až po reklamní účely, kde uživatelé ocení malou náročnost na přenesená data při zachování kvality snímků.

5.1.1 Prohlížení fotografií s vysokým rozlišením

Klasickým příkladem je prohlížečka fotografií s vysokým rozlišením. Takovéto fotografie lze velmi rychle načíst a přibližovat. Díky pružným animacím a speciálním úpravám snímku lze prohlížet i Giga-pixelové fotografie v reálném čase, což klasické prohlížečky neumožňují, jelikož načtení, respektive přiblížení může trvat několik desítek sekund.

5.1.2 Panoramatické fotografie

Vyfocením série snímků, které spolu tvoří obraz místnosti, který zabírá zorný úhel 360° vznikne panoramatická fotografie, kterou lze taktéž převést do formátu DeepZoom. Takto upravenou fotografii může uživatel libovolně posouvat a přibližovat, nabývá tak dojmu, že se nachází v dané místnosti.

5.1.3 Reklamní využití

Vytvořením obrázku s relativně nízkým rozlišením, který reprezentuje vzhled reklamy. Následnou uživatelskou interakcí jsou načítány další úrovně, které obsahují více údajů o nabízeném produktu. Vzhledem k postupnému načítání obrázku nedochází ke zbytečnému zatížení hostitelského webu a na relativně malé ploše je možno zobrazit velké množství informací.

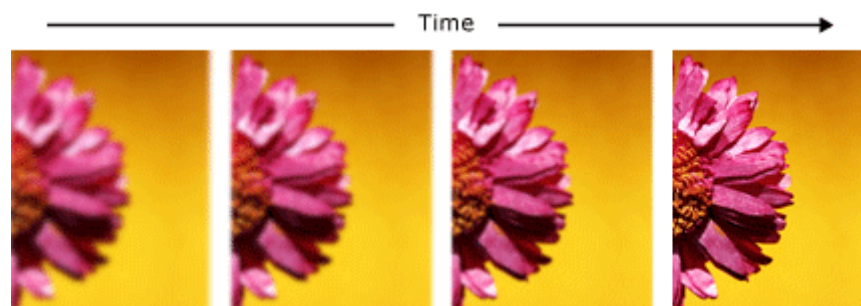
5.2 Historie

Technologie je velmi podobná Google mapám, kde jsou snímky rozděleny do vrstev, které jsou zobrazovány podle potřeby. Hlavním rozdílem je že, Google mapy využívají pro každou vrstvu jinou sérii snímků, kdežto DeepZoom pracuje na každé vrstvě se stejným snímkem o jiném rozlišení.

5.3 Funkce

DeepZoom využívá snímky s různou úrovní rozlišení a postupným načítáním pro dosažení rychlého otevření velmi velkých snímků. Při načítání stránky je zapotřebí jen malé množství dat pro zobrazení požadovaného obrázku, který může uživatel vidět takřka okamžitě. V počátečních chvílích obsahuje obrázek jen velmi nízké rozlišení, jež se postupem času, tak jak je obrázek načítán, zlepšuje do doby, než jsou k dispozici všechna data. To je ten důvod, proč můžeme pozorovat přechod od rozostřeného k ostrému obrázku.

DeepZoom využívá pružné animace, které dodávají uživatelům dojem plynulého pohybu, ať už při posouvání obrazu nebo jeho přibližování. Tyto animace jsou spouštěny při jakékoliv uživatelské interakci. DeepZoom plně podporuje hardwarovou akceleraci.



Obrázek 19: Časová osa načtení DeepZoom obrázku (obrázek převzat, zdroj č. 31).

5.4 Dělení

DeepZoom využívá souborový systém, který je založen na XML. V tomto kódu může uživatel specifikovat formát konečného obrázku. Jedná se o samostatný obrázek o vysokém rozlišení nebo kolekci menších obrázků, jež spolu tvoří jeden celek.

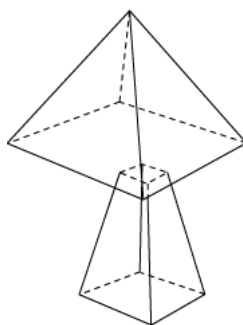
5.4.1 Samostatný obrázek

Samostatné obrázky o vysokém rozlišení jsou v DeepZoom reprezentovány jako dlaždicová pyramida. To umožňuje renderovacímu systému přenést a zobrazit jen určitou část, jenž je třeba zobrazit. Je-li obraz oddálen, poté stačí přenést jen malý náhled o nízkém rozlišení.

Nicméně pokud se uživatel rozhodne obraz přiblížit určitou oblast obrazu, potom renderovací systém přenese jen ty dlaždice, které jsou potřeba pro vyplnění aktuálního zobrazení. Tento fakt má velký vliv na datový přenos.

5.4.2 Rozptýlený obrázek

Deepzoom podporuje koncept rozptýlených obrázků. Rozptýlené obrázky umožňují uživateli vytvořit Obrázek, který obsahuje na určitých místech vyšší rozlišení. To je vhodné u reklamního letáku, kde prodejce inzeruje nový typ automobilu a v pravém dolním rohu leták obsahuje vyšší rozlišení s doprovodným textem. Zájemce si poté může leták přiblížit a doprovodný text snadno přečíst.



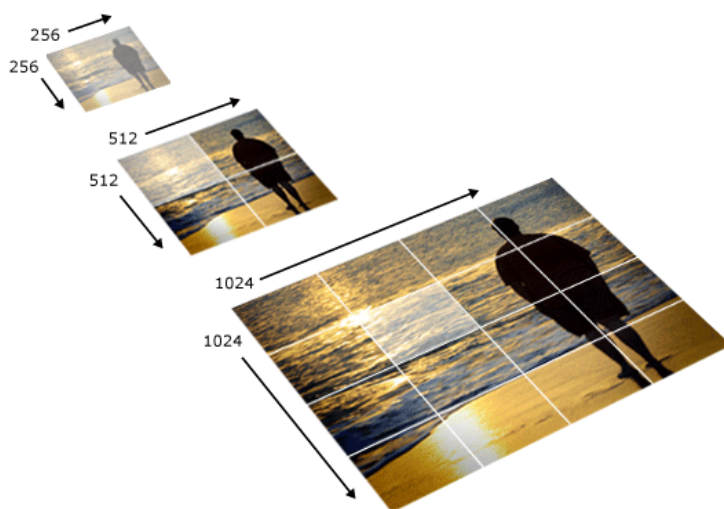
Obrázek 20: Rozptýlený DeepZoom obrázek (obrázek převzat, zdroj č.32).

5.4.3 Kolekce Obrázků

Kolekce je složena z více obrázků. DeepZoom optimalizuje práci s touto kolekcí tak, že umožňuje sdílet nižší úroveň obrázků do jedné společné dlaždice a načíst tak stovky obrázků s vysokým rozlišením v jednom procesu. Po přiblížení na určitý obrázek, jsou už načítány dlaždice, jež odpovídají konkrétnímu snímku.

5.5 Vytvoření DeepZoom Obrázku

DeepZoom obraz je složen z dlaždic ve formátu JPEG nebo PNG v různých rozlišeních, které tvoří obraz pyramidy. Velikost dlaždice je typicky 256 X 256 pixelů, ale tuto velikost lze upravit. Každá dlaždice je uložena v samostatném souboru a každá úroveň pyramidy je uložena v samostatné složce. To umožňuje aplikaci DeepZoom přenést a načíst, pouze ty dlaždice, které jsou potřeba pro aktuální velikost obrazu. Například pokud přiblížíme střední část obrazu DeepZoom načte pouze vyznačené dlaždice namísto celého obrazu 1024 X 1024 pixelů.



Obrázek 21: DeepZoom pyramida (obrázek převzat, zdroj č. 33).

Ruční tvorba těchto pyramid je velmi náročná jak na čas, tak i na přesnost, proto je doporučován Nástroj DeepZoom Composer. Tento nástroj umožňuje vytvoření požadované pyramidy z jednoho nebo několika obrázků s vysokým rozlišením. Pro přístup k pyramidě se používá XML schéma, které je součástí vygenerované kompozice.

5.6 Načtení obrázku:

V silverlightu je DeepZoom zastoupen objektem MultiscaleImage

Takto vypadá přidání objektu v XAML:

```
<MultiScaleImage x:Name="deepZoomObject" Source='ferrari/dzc_output.xml' />
```


A takto v code-behind (C#):

```
MultiScaleImage myDeepZoomObject = new MultiScaleImage();  
myDeepZoomObject.Source = new Uri("ferrari/dzc_output.xml");
```

Po načtení ovšem uživatel nemá přístup k jakékoli funkci. To lze napravit přidáním události **MouseEnter** objektu **myDeepZoomObject**

XAML:

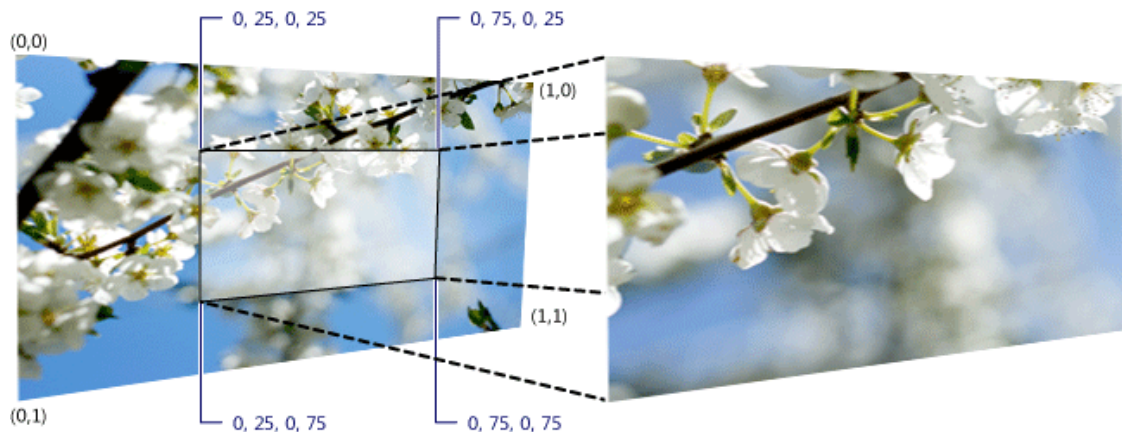
```
<MultiScaleImage x:Name="mydeepZoomObject" Source='ferrari/dzc_output.xml'  
    MouseEnter="DeepZoomObject_MouseEnter" />
```

Událost v C#:

```
private void DeepZoomObject_MouseEnter(object sender, MouseEventArgs e)  
    // ZoomAboutLogicalPoint je metoda která umožňuje přiblížení  
    // První parametr je hodnota přiblížení (3x)  
    // druhý a třetí parametr reprezentují osu X a Y v souřadnicovém systému  
    // 0..1  
    this.deepZoomObject.ZoomAboutLogicalPoint(3, .5, .5);  
}
```

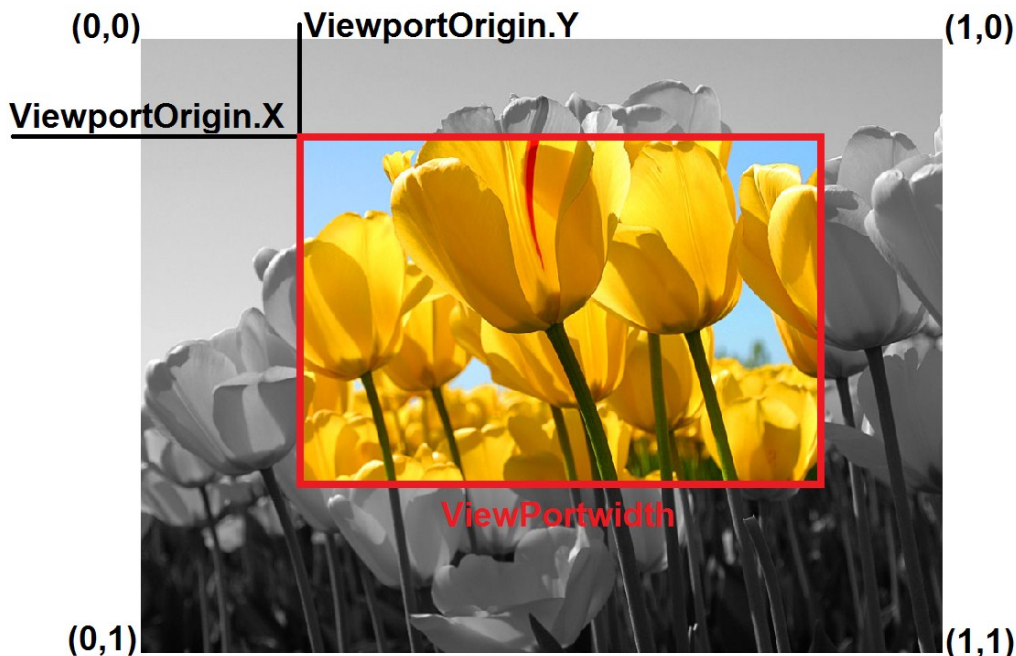
Metoda **ZoomAboutLogicalPoint** umožňuje zvětšovat a posouvat obrázek. První parametr je hodnota kolikrát má být obrázek zvětšen. Například v tomto příkladu bude obrázek přiblížen třikrát. Po opětovném zavolání události bude obrázek přiblížen opět třikrát, tedy 9 krát oproti původní velikosti. Pro oddálení obrázku je nutno použít hodnotu, která je ostře menší než 1. Například hodnota 0,33 oddálí obrázek přibližně 3X

Druhý a třetí parametr metody **ZoomAboutLogicalPoint** jsou příslušné osy X a Y, jež určují na které místo, má být obrázek přiblížen. Logický bod používá normalizované hodnoty 0 až 1 kde 0 je počátek obrázku a 1 jeho konec jak ve vodorovném směru tak i v horizontálním proto použité hodnoty 0,5,0,5 přiblíží obraz přesně na střed.



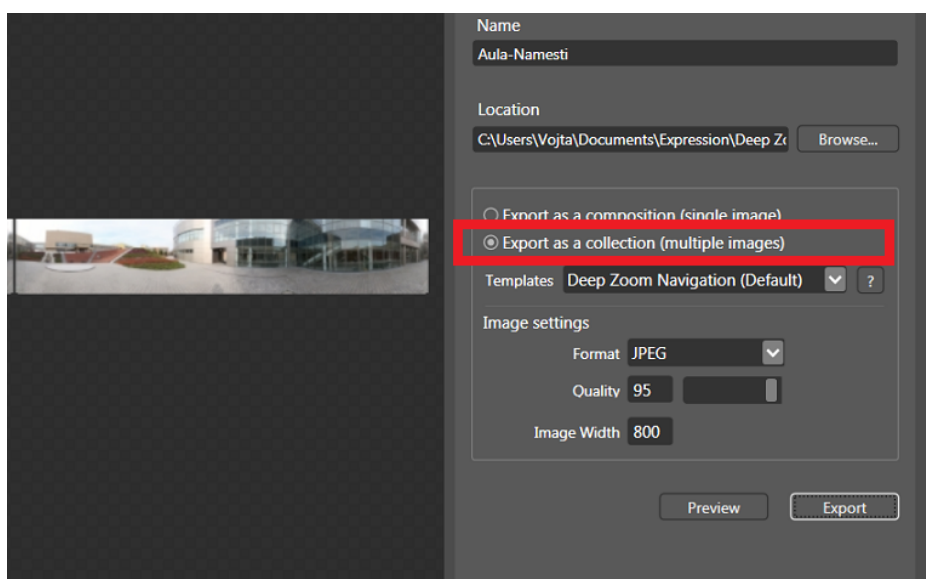
Obrázek 22: Přiblížení DeepZoom obrázku (obrázek převzat, zdroj č. 34).

Mezi důležité funkce patří také metoda **ViewportOrigin** respektive **ViewportWidth**. První jmenovaná metoda nastavuje nebo vrací souřadnice levého horního rohu aktuálně zobrazovaného okna vzhledem k normalizované velikosti (0 až 1). Metoda **ViewportWidth** pak udává šířku aktuálně zobrazovaného okna. To znamená, že čím menší tato hodnota je, tím více je DeepZoom obrázek přiblížen. Bohužel komponenta **MultiscaleImage** nemá neimplementovanou metodu, která by vracela výšku zobrazovaného okna, což může negativně ovlivnit práci s touto komponentou.



Obrázek 23: Znárodnění prohlížení DeepZoom obrázku.

Jak již bylo dříve zmíněno DeepZoom umožňuje práci s kolekcí obrázků. Umožňuje to property **SubImages** komponenty **MultiscaleImage**. Práce s touto kolekcí je velmi jednoduchá, jediným kritériem je zvolit volbu „export as a Collection (multiple images)“ při exportu obrázku pro DeepZoom.



Obrázek 24: Export obrázku jako kolekce.

Přístup k jednotlivým obrázkům se provádí pomocí indexů.

```
mydeepZoomObject.SubImages[0].ViewportOrigin.X
```

Každý obrázek kolekce disponuje property `ViewportOrigin` a `ViewporWidth`, jež nastavuje jeho pozici a velikost. Celá kolekce se pak chová jako samostatný obrázek, s nímž lze pohybovat a přibližovat jej. Nepříjemnou vlastností jednotlivých obrázků kolekce je, že tyto obrázky již nedisponují žádnou událostí. Tudíž nelze rozeznat, na který obrázek uživatel zrovna klikl a podobně.

Další nejpoužívanější metody a události, které lze použít při práci s komponentou `MultiscaleImage` jsou:

<i>ActualHeight</i>	- Poskytuje aktuální výšku
<i>ActualWidth</i>	- Poskytuje aktuální šířku
<i>UseSprings</i>	- Umožňuje dočasně pozastavit používání pružných animací
<i>ZoomAboutLogicalPoint</i>	- Umožňuje přiblížit obrázek na určité místo

6 Aplikace pro virtuální prohlídky

Cílem této práce bylo také vytvořit aplikaci, která bude umožňovat virtuální prohlídky sférických panoramat za použití technologie Microsoft Silverlight a DeepZoom. Důraz je kladen na datovou nezávislost. To znamená, že zobrazovaný obsah musí být jednoduchým způsobem lehce změnitelný. Další podmínkou byla snadná implementace prohlížečky do dalších aplikací.

6.1 Analýza

Aplikace musí umožňovat posouvání snímků a jejich přibližování respektive oddalování na konkrétní místo pro prozkoumání jednotlivých detailů.

Dalším požadavkem je vytvoření takzvané iluze nekonečného obrázku. To znamená, že uživatel musí nabývat dojmu, že může panorama stále posouvat bez toho, aniž by někdy došel k levému nebo pravému konci snímku. Tato funkce umožňuje vytvořit iluzi, že se uživatel v panoramatu otáčí kolem své vlastní osy. Jelikož se jedná o panoramatickou fotografii, která zabírá zorný úhel 360° horizontálně, tak při posunu fotografie vzniká iluze, že se uživatel otáčí, tak jako by byl skutečně na místě, které panorama zachycuje.

Následujícím požadavkem je zajištění přechodu mezi jednotlivými panoramaty v rámci jedné aplikace, tak aby uživatel mohl lehce přecházet z jednoho snímku do druhého, pomocí uživatelské interakce.

Posledním úkolem je obohatit virtuální prohlídku o informace, jež popisují nebo doplňují jednotlivé objekty, které se nacházejí v panoramatu. Například pokud panorama zachycuje školní třídu, kde na zdi visí nástěnka s rozvrhem hodin, musí mít uživatel možnost, pomocí interakce s tímto rozvrhem hodin, přejít na webovou adresu, kde se nachází konkrétní rozvrh hodin s doplňujícími informacemi o jednotlivých hodinách.

6.2 Návrh

Na základě analýzy zadání aplikace, byl proveden návrh řešení, které zahrnuje posun a přiblížení snímku, vytvoření iluze nekonečného snímku a přechod na jiný snímek v rámci jedné aplikace, pomocí hotspotu, které zároveň slouží i jako nástroje, jež umožňují zobrazit doplňkové informace o objektech v panoramatu.

Inspirací při návrhu aplikace byly ukázkové aplikace Microsoft Silverlight na adrese:

<http://samples.msdn.microsoft.com/silverlight/samplebrowser/index.htm#/?sref=DeepZoomFullSample>

6.2.1 Posun snímku

Je navržen tak, že uživatel klikne pomocí myši nebo pomocí jakéhokoliv jiného polohovacího zařízení na obrazovku a za stálého držení tlačítka posouvá snímek na libovolnou stranu, tak jak uživatel pohybuje kurzorem na obrazovce. Po uvolnění tlačítka myši dojde k ukončení posunu snímku. Toto řešení vychází z reálného prostředí, například v modelové situaci kde jsou na stole rozmístěny fotografie a uživatel je chce posunout na jiné místo, musí nejprve na konkrétní fotografii položit prst, což je v případě aplikace klinutí tlačítka myši, a za stálého držení fotografii posouvá po stole, do té doby než se fotografie nachází na požadovaném místě a v tomto okamžiku prst zvedne, což je opět v aplikaci simulováno pomocí uvolnění tlačítka myši.

6.2.2 Přiblížení

Přiblížení snímku je navrženo dvěma způsoby, prvním způsob je přiblížení pomocí kolečka myši. Jedná se o pohodlný a hlavně rychlý způsob jak obrázek přiblížit nebo oddálit. Druhým způsobem je ovládací prvek, jenž se nachází v aplikaci. Tento druhý způsob je navržen proto, že při prohlížení panoramatu na notebooku bez použití myši je k dispozici touchpad, který umožňuje snímek posouvat, ale už se zde většinou nenachází žádné rolovací kolečko, které by přibližování umožňovalo. Proto aplikace obsahuje ovládací prvek, který přiblížení respektive oddálení umožňuje i bez použití myši.

6.2.3 Iluze nekonečného obrázku

Nekonečný snímek je navržen tak, že jsou na plochu nahrány dva totožné snímky vedle sebe, přičemž uživatel, vidí pouze jediný snímek. Jelikož se jedná o snímky, které zabírají zorný úhel 360° tak na sebe plynule navazují, ať už na pravé nebo levé straně. Uživatel tedy nepozná, že přešel z jednoho snímku na druhý. Navíc jsou jednotlivé snímky přesouvány tak, jak se uživatel po panoramatu pohybuje. V okamžiku kdy se uživatel blíží k levému okraji snímku číslo jedna, je před tento snímek přesunut snímek číslo dvě a analogicky pokud se uživatel blíží k pravému okraji snímku dvě, je před tento snímek přesunut snímek číslo jedna a uživatel na tento snímek plynule přejde.

6.2.4 Přechod na jiný snímek

Pro přechod na jiný snímek byly navrženy tzv. Hotspoty. Jedná se o body, které jsou svázány s konkrétním místem na panoramatu. To znamená, že v okamžiku kdy uživatel pohybuje s panoramatem tak pohybuje zároveň i s těmito hotspoty a při jejich interakci dojde k předdefinované akci, například přechod na nový snímek nebo zobrazení doplňujících informací o objektech, které se nacházejí v tomto panoramatu.

6.3 Implementace

V této kapitole se nachází detailní popis implementace předchozích návrhů řešení aplikace umožňující virtuální prohlídky. Je zde popsáno přiblížení a pohyb snímku, dále implementace nekonečného obrázku a vytvoření hotspotu s problémy a jejich řešením, které vytvoření hotspotu provází.

6.3.1 Přiblížení

Přiblížení snímku je prováděno pomocí metody `ZoomAboutLogicalPoint` komponenty `MultiscaleImage`, kde první parametr je hodnota zvětšení (hodnota je větší než jedna), nebo zmenšení snímku (hodnota je menší než jedna) další dva parametry jsou osa X a osa Y. Tyto souřadnice určují bod, na který má být snímek přiblížen. Zde je názorná ukázka:

```
this.ZoomImage.ZoomAboutLogicalPoint(zoom / newzoom, logicalPoint.X,  
    logicalPoint.Y);
```

V ukázkové aplikaci je přiblížení respektive oddálení realizováno pomocí ovládacích prvků číslo 4 a 5. (obrázek číslo) Dále lze použít kolečko myši. Komponenta `MultiscaleImage` disponuje událostí **MouseWheel**, která přiblížování a oddalování pomocí kolečka myši zajišťuje. Problémem je, že při testování aplikace v různých prohlížečích docházelo k nestandardnímu chování této události. Proto byla převzata událost třetí strany, která toto nestandardní chování eliminovala. Tuto událost lze nalézt na adrese:

http://samples.msdn.microsoft.com/silverlight/samplebrowser/index.htm#/?sref=DeepZoomSimpleZoom_InOnly

6.3.2 Pohyb

Pohyb snímku je prováděn pomocí metody `ViewportOrigin` komponenty `MultiscaleImage`, která umožňuje vrátit nebo nastavit levý horní roh aktuálně zobrazovaného snímku vzhledem k normalizovaným souřadnicím X a Y.

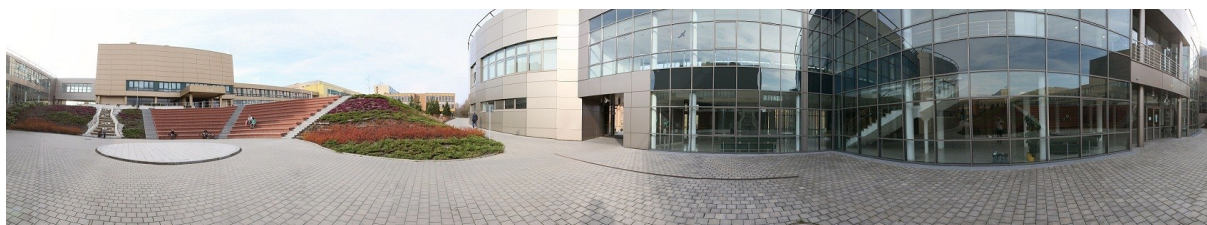
Konkrétně tak, že k původní souřadnici X je přičítán rozdíl souřadnic myši při zmáčknutí tlačítka a následném pohybu myši. Tento rozdíl je pak vydělen součinem Celkové šířky a aktuální šířky zobrazovaného okna v normalizovaném tvaru. Stejně je postupováno u osy Y.

Zde je názorná ukázka kódu, který přiblížení zajišťuje:

```
Point newPoint = lastMouseViewport;  
newPoint.X += (lastMouseLogicalPos.X - thisMouseLogicalPos.X) /  
this.ZoomImage.ActualWidth * this.ZoomImage.ViewportWidth;  
newPoint.Y += (lastMouseLogicalPos.Y - thisMouseLogicalPos.Y) /  
this.ZoomImage.ActualWidth * this.ZoomImage.ViewportWidth;  
this.ZoomImage.ViewportOrigin = newPoint;
```

6.3.3 Iluze nekonečného obrázku

Problém nekonečného obrázku spočívá v tom, že uživatel při prohlídce panoramatu může pohybovat snímkem stále dokola, aniž by se někdy dostal k jeho konci. Toto je problém, jelikož aplikace Microsoft ICE, která slouží ke generování panoramat z jednotlivých snímků poskytuje následující pohled:



Obrázek 25: Nekonečný obrázek krok 1.

Je zřejmé, že při pohybu snímkem uživatel dojde k jeho levému nebo pravému konci. Tento problém je řešen tak, že jsou do komponenty MultiscaleImage vloženy tyto snímky dva pomocí kolekce, jež tato komponenta podporuje. Po těchto úpravách vypadá snímek následovně:



Obrázek 26: Nekonečný obrázek krok 2.

Tímto daný problém, však stále nekončí, jelikož při pohybu snímkem stejně uživatel dojde k levému nebo pravému konci snímku. Řešení spočívá v tom, že jednotlivé snímky jsou přehazovány podle toho, v jaké pozici se nachází okno, které vidí uživatel.



Obrázek 27: Nekonečný obrázek krok 3.

Červený rámeček (obrázek č. 27) naznačuje prostor, který aktuálně uživatel vidí, tato velikost je pevně určena a odpovídá přibližně zornému poli člověka. Při posunu snímku jsou kontrolovány souřadnice levého horního rohu rámečku a při překročení kritické meze je snímek při posunu vpravo (uživatel se dívá doleva) dojde k přesunu snímku číslo dvě před snímek číslo jedna. Tak aby při dalším posunu celého panoramatu uživatel plynule přešel na snímek číslo dvě.



Obrázek 28: Nekonečný obrázek krok 4.

Toto řešení je možné díky komponentě MultiscaleImage, která může obsahovat kolekci obrázků, kde velikost celé kolekce odvozena od normalizovaného souřadnicového systému. Levý horní roh kolekce má tedy souřadnice (0,0) a pravý horní roh souřadnice (1,0). U obrázku kolekce je situace následující: levý horní roh obrázku číslo jedna má souřadnice (0,0), pravý horní roh má souřadnice (0.5,0) a levý horní roh obrázku číslo dvě má souřadnice levého respektive pravého horního rohu

(0.5,0) respektive (1,0). Při posunu obrázku číslo dvě před obrázek číslo jedna je jeho hodnota ViewPortOrigin.X nastavena na hodnotu -0.5, souřadnice obrázku číslo jedna nyní bude (-0.5,0)

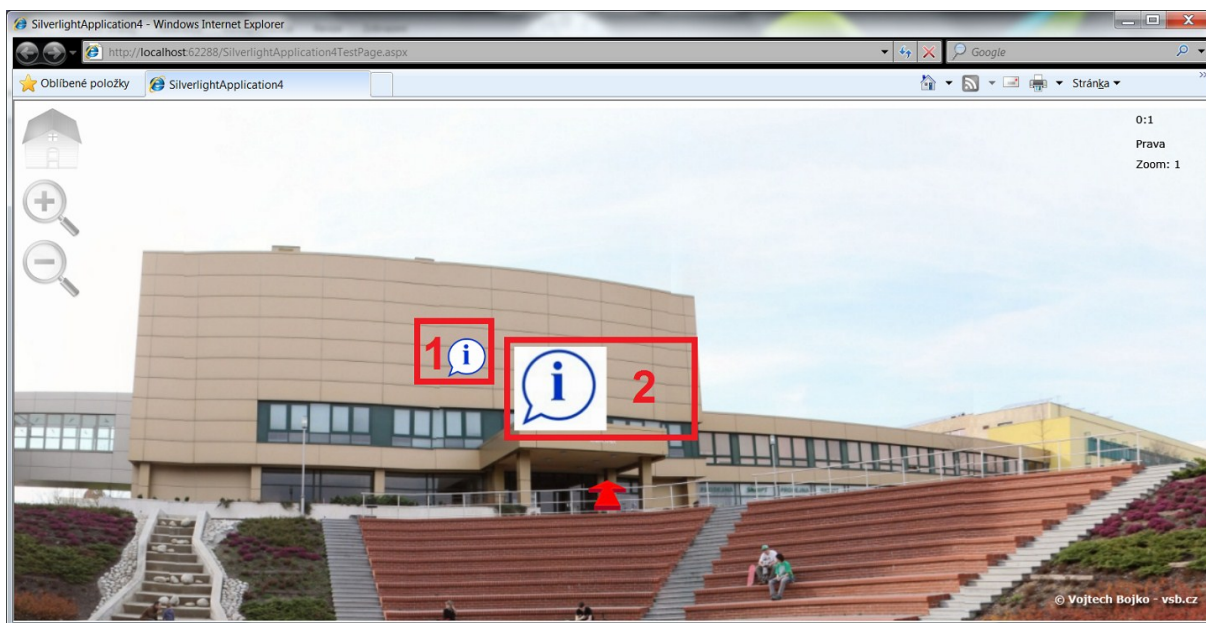
```
msi.SubImages[1].ViewportOrigin.X=-0.5;
```

6.3.4 Vytvoření hotspotu

Původní záměr byl, využít kolekci obrázků, jež komponenta MultiscaleImage poskytuje z důvodu, že tento hotspot by se automaticky pohyboval tak, jak by bylo pohybováno s celým panoramatem. Hotspot by byl jen dalším obrázkem v kolekci obrázků komponenty MultiscaleImage. Avšak v průběhu implementace byl tento záměr zamítnut a to hned ze dvou závažných důvodů.

Prvním důvodem bylo to, že hotspot má plnit pouze doplňkovou a informativní činnost panoramatu. Při použití kolekcí menších DeepZoom obrázků docházelo k tomu, že se tyto obrázky zvětšovaly a zmenšovaly tak, jak uživatel panoramatický snímek přibližoval nebo oddaloval.

V okamžiku kdy uživatel panorama několikanásobně přiblížil, k čemuž je DeepZoom primárně určen, zabíraly tyto hotspoty celou šíři obrazovky a značně znepříjemňovali prohlížení panoramatického snímku. Tak, jak to lze vidět na obrázku číslo 29 a 30 v rámečku číslo dvě.



Obrázek 29: Problém hotpotu část 1.

- 1) Hotspot realizován komponentou Image
- 2) Hotspot realizován jako kolekce komponenty MultiScaleImage

Proto byly zavedeny takové hotspoty, které při jakémkoli přiblížení nebo oddálení zabírají stejnou šíři obrazovky uživatele. Tyto hotspoty lze vidět na obrázku 29 a 30 rámeček číslo jedna.



Obrázek 30: Problém hotspotu část 2.

- 1) Hotspot realizován komponentou *Image*
- 2) Hotspot realizován jako kolekce komponenty *MultiScaleImage*

Druhým závažným důvodem, bylo to, že jednotlivé obrázky v kolekci komponenty *MultiScaleImage* neobsahují žádnou událost. Tudíž při kliknutí na konkrétní obrázek nelze vyvolat jakoukoli interakci. Tak, jak je to například u samostatných *DeepZoom* obrázků.

Z těchto důvodů byl návrh využití kolekce obrázku zamítnut a bylo použito následující řešení, které spočívá v tom, že přes celou komponentu *MultiScaleImage* je natažena komponenta *canvas*. Tento *canvas* následně obsahuje jednotlivé hotspoty reprezentované komponentou *image*.

Toto řešení však sebou nese několik problému, které by se v případě použití první varianty nevyskytovaly. Jedná se o synchronizaci hotspotu s *DeepZoom* obrázkem a synchronizaci hotspotu s nekonečným obrázkem.

Synchronizace hotspotu s *DeepZoom*

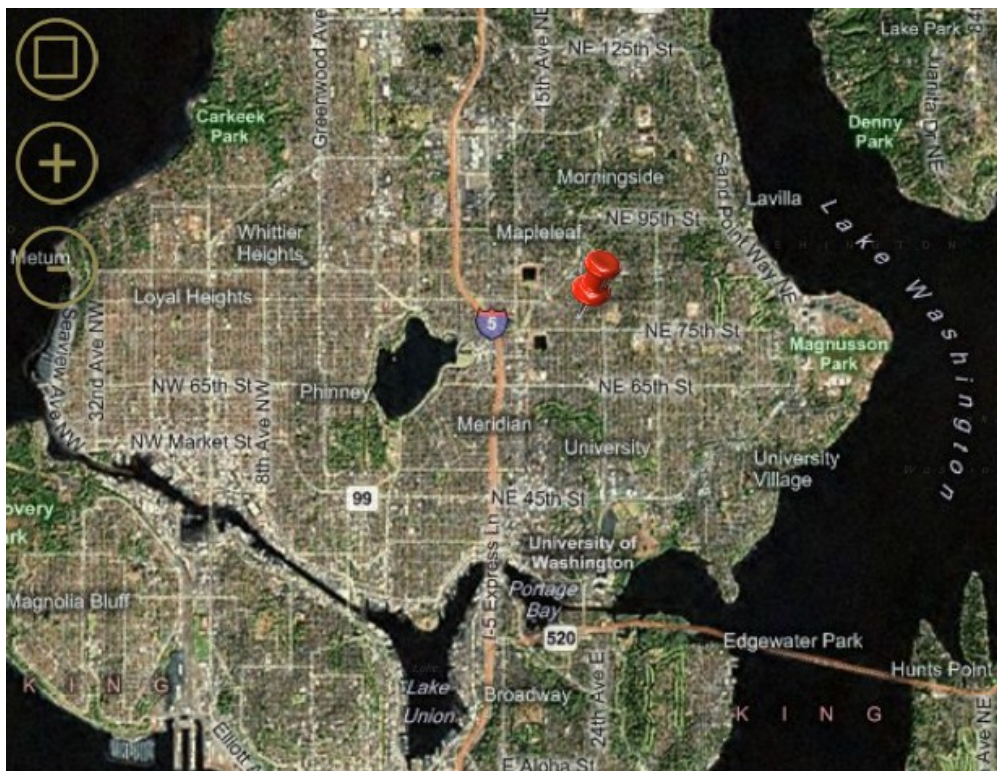
Synchronizace hotspotu a komponenty *MultiScaleImage*, vyžaduje několik nutných kroků. Nejprve musí být přepočteny souřadnice komponenty *canvas*, které mají obvykle rozměr okna webového prohlížeče na rozměry komponenty *MultiScaleImage*, která pracuje s normalizovaným souřadnicovým systémem 0 až 1.

Je nutné, aby se komponenta *image* v komponentě *canvas* pohybovala tak, jak pohybujeme s panoramatem v komponentě *MultiScaleImage*. Dále je nutné udržovat stále stejnou velikost komponenty *image*. Bez ohledu na to zda je panorama přiblíženo, nebo oddáleno.

Dalším problémem je skutečnost, že pohyb nebo přiblížení *DeepZoom* snímku je doprovázeno pružnou animací, která nastává po uvolnění levého tlačítka myši a obrázek „doplouvá“ ještě o kus dál než jsou souřadnice v době uvolnění tlačítka myši. Tento pohyb musí také vykonávat komponenty *image* v komponentě *canvas*.

Pro synchronizaci jsou použity metody třetích stran, původně určené k synchronizaci map, které se nacházely v komponentě MultiscaleImage, a ukazatelů. Tyto ukazatele označovaly určitá místa na této mapě. Původní aplikaci lze nalézt na adrese:

<http://blogs.msdn.com/b/lutzg/archive/2008/08/19/synchronizing-images-with-the-deep-zoom-content.aspx>



Obrázek 31: Aplikace třetí strany (obrázek převzat, zdroj č. 31).

Synchronizace hotspotu s „nekonečným obrázkem“

Posledním problémem je synchronizace nekonečného obrázku s hotspotem. Nekonečný obrázek zajišťuje to, že se uživatel může v panoramatu stále otáčet, aniž by došel k levému nebo pravému okraji snímku. Problém je řešen přehazováním snímku podle toho kam se uživatel právě dívá. S tím souvisí problém hotspotu a panoramatu. Faktem je, že je hotspot napevno přichycen k určitému bodu normalizovaného souřadnicového systému, kdežto jednotlivé obrázky DeepZoom kolekce jsou neustále posouvány podle toho, jak uživatel panorama posouvá.

Řešením je změna souřadnice X daného hotspotu. To znamená, že pokud hotspot překročí pravý okraj obrazovky je přesunut doleva a analogicky pokud překročí levý okraj obrazovky je přesunut doprava.

6.3.5 Načítání hotspotu

Jednotlivé hotspoty jsou dynamicky načítány z XML souboru. To umožňuje uživatelům přidávat hotspoty do panoramat přímo za běhu aplikace. Každý hotspot je reprezentován elementem s názvem image a každý element obsahuje atributy, jež určují, na kterém panoramatu bude daný hotspot zobrazen. Dále každý element obsahuje stejnou sadu tagů, které obsahují řadu informací jako například souřadnice X a Y, na které určují v jakém místě se má hotspot nacházet.

Schéma XML souboru vypadá následovně:

```
<?xml version="1.0" encoding="utf-8" ?>
<page name="Stranka">
  <Panorama name="panorama1">
    <Image name="1">
      <URL>/SilverlightApplication4;component/arrow.png</URL>
      <X>0,09</X>
      <Y>0,030649</Y>
      <Hint> goto </Hint>
      <Navi>Yes</Navi>
      <WWW>www</WWW>
      <DeepzoomUrl number="2" X="0,1873" Y="0,02245">Hrad/dzc_output.xml</DeepzoomUrl>
    </Image>
    <Image name="1">
      <URL>/SilverlightApplication4;component/informace.png</URL>
      <X>0,3275</X>
      <Y>0,02361</Y>
      <Hint>info </Hint>
      <Navi>No</Navi>
      <WWW>http://cs.wikipedia.org/wiki/V%C4%8Del%C3%AD_%C3%BA1</WWW>
      <DeepzoomUrl number="0" X="0" Y="0">NULL</DeepzoomUrl>
    </Image>
  </Panorama>
</page>
```

Tento XML soubor obsahuje dva hotspoty, kde první je navigační a druhý je informativní. Každý hotspot je uložen v samostatném elementu `<Image>`. Každý element obsahuje atribut `name`, který určuje, na kterém panoramatu má být zobrazen. Každý element dále obsahuje stejnou sadu tagu.

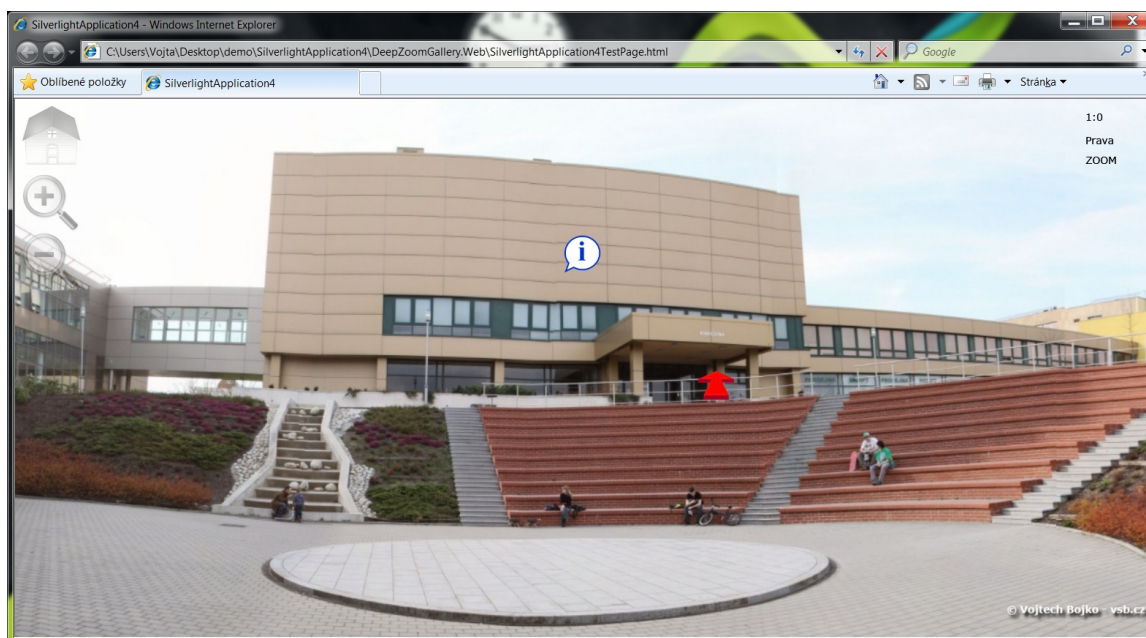
Popis tagů

- `<URL>` Obsahuje url adresu obrázku, který hotspotu přiřazuje jeho vzhled.
- `<X>` Souřadnice X normalizovaného souřadnicového systému, kde má být hotspot zobrazen.
- `<Y>` Souřadnice Y normalizovaného souřadnicového systému, kde má být hotspot zobrazen.
- `<Hint>` Obsahuje popis hotspotu, který se objeví po najetí kurzoru na hotspot. Tato funkce však ještě není doimplementována.
- `<Navi>` Udává, zda se jedná o navigační nebo informativní hotspot. V případě, že se jedná o navigační hotspot načte se nové panorama z url adresy, který se nachází v tagu.
- `<DeepzoomUrl>`. V opačném případě se otevře nové okno prohlížeče s http adresou, která se nachází v tagu `<WWW>`.
- `<WWW>` Obsahuje http adresu, která se načte po kliknutí na informační hotspot.

- **<DeepzoomUrl>** Obsahuje url adresu panoramatu, která se načte po kliknutí na navigační hotspot. Tento tag obsahuje tři atributy: **number**, **X** a **Y**. První atribut určuje, které hotspoty se na tomto panoramatu mají zobrazovat. V Ukázce atribut obsahuje číslo 2. To znamená, že všechny hotspoty s atributem **name**, který obsahuje číslo 2, budou na tomto panoramatu zobrazeny. Druhý a třetí atribut určuje souřadnice, jež nastavují výchozí bod při načtení nového panoramatu. Tento výchozí bod je důležitý při přecházení z jednoho panoramatu do druhého. V okamžiku kdy uživatel klikne na navigační hotspot je přenesen do nového panoramatu a výchozí bod zajišťuje to, aby se uživatel díval pořád stejným směrem i v novém panoramatu.

7 Navržená aplikace

Navržená aplikace umožňuje prohlížení 360 stupňových panoramat, které jsou vytvořeny pomocí nástroje DeepZoom Composer. Využívá kolekci panoramat pro vytvoření iluze nekončícího snímku, v němž se může uživatel pohybovat (otáčet) tak, jako by stál přímo na zobrazovaném místě, ať už se jedná o místnost nebo zajímavou vyhlídku. Jednotlivé pohledy jsou navíc mezi sebou propojeny pomocí hotspotů, které jsou zobrazovány přímo v panoramatu a umožňují měnit právě zobrazovaný obsah v reálném čase.



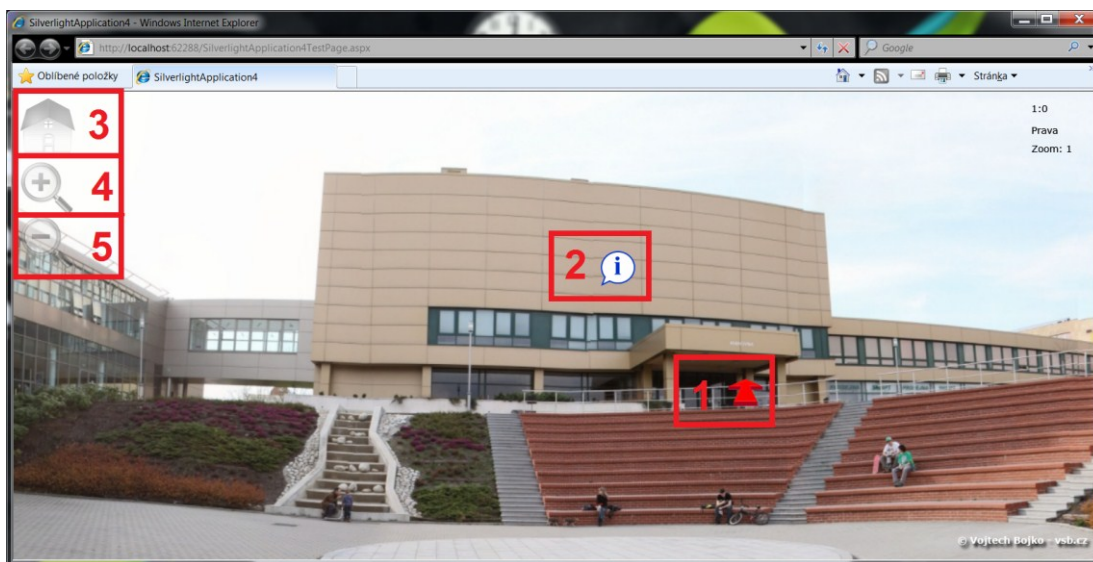
Obrázek 32: Navržená aplikace

Při jejich interakci dochází k předem definované události, například požadovaný přechod na jiný pohled. Tyto hotspoty jsou dynamicky načítány ze souboru XML. Uživatel jsi tedy do určité míry, může navolit, kde a jaké hotspoty mají být zobrazovány. V aplikaci jsou použity dva základní druhy:

- Navigační hotspot, který umožňuje při kliknutí načíst jiné panorama a tím se přesunout na jiné místo.
- Informativní hotspot, který při kliknutí načte internetovou stránku, jež může obsahovat informace o daném předmětu, jež se nachází v panoramatu a na který hotspot poukazuje.

7.1 Funkce

Prohlížečka obsahuje základní funkce, které uživatelům umožňují příjemné prohlížení panoramat. Jedná se především o pohyb (otáčení) v panoramatu, přiblížení detailu a přechod na další panorama respektive zobrazení detailní informace o předmětech, za použití hotspotů.



Obrázek 33: Funkce ukázkové aplikace.

- 1) Navigační hotspot, po kliknutí dojde k přesunu na jinou pozici.
- 2) Informativní hotspot, po kliknutí dojde k otevření okna prohlížeče s navolenou internetovou adresou.
- 3) Ovládací prvek, jenž vrátí obrazovku do původní pozice.
- 4) Ovládací prvek umožňující přiblížit snímek na střed.
- 5) Ovládací prvek umožňující oddálit snímek.

7.2 Využití

Aplikace má široké využití, ať už z pohledu veřejného sektoru, kde mohou uživatelé procházet jednotlivé části města, bez toho aniž by překročili vlastní práh domu či bytu. V komerčním sektoru lze využít aplikaci pro prezentování staveb, zahrad nebo bytů. Velký potenciál pro využití je u realitních kanceláří, kde si klienti mohou prohlédnout vybraný byt, bez toho aniž by ho fyzicky navštívili. Což přináší značné úspory jak z finanční stránky, tak z pohledu časové náročnosti. Díky tomu že, je aplikace navržena pomocí technologie Microsoft Silverlight může být aplikace snadno implementována na jakoukoli webovou stránku, která splňuje požadavky pro technologii Silverlight.

7.3 Další vývoj

Jak už bylo zmíněno, snahou je uplatnit aplikaci jak v komerčním, tak ve veřejném sektoru. Problémem je, že uživatel bez znalosti problematiky není schopen panorama sám bez pomoci vyrobit. Další vývoj by se měl ubírat tak, aby uživatel byl schopen vyrobit vlastní panorama pouze se základní znalostí problematiky, konkrétněji pouze to, jak mají být jednotlivé snímky vyfoceny a zpracovány.

Závěr

Virtuální prohlídky jsou bezpochyby dalším evolučním krokem, jak zachytit panorama a zvěčnit jej pro další generace v digitální podobě. K tomuto účelu je k dispozici řada nástrojů. Námi použité technologie Microsoft Silverlight a DeepZoom mají oproti svým konkurentům jednu zásadní výhodu a tou je rychlost s jakou jsou fotografie načítány a zobrazovány. Když bereme v úvahu, že pracujeme mnohdy i s Giga-pixelovými fotografiemi, tak je tato rychlost obdivuhodná a spolu s pružnými animacemi, které prohlížení panoramat doplňují, tvoří příjemný zážitek z plynulého prohlížení virtuálních prohlídek.

V první části dokumentu je shrnuto, jak vytvořit fotografii vhodnou pro virtuální prohlídku, která je kompatibilní s technologií DeepZoom. Nachází se zde několik rad, jak požadovanou fotografii vytvořit, na co si dát pozor a čeho se vyvarovat.

V druhé části je uveden popis použitých technologií .Net, Silverlight a DeepZoom. Další významnou částí tohoto dokumentu je nastínění postupu, jak vytvořit aplikaci která virtuální prohlídky za použití nástrojů Microsoft Silverlight a DeepZoom umožňuje. Dále jsou zde uvedeny i problémy a jejich řešení, které jsou klíčové pro vytvoření této aplikace.

Další budoucí vývoj projektu virtuální prohlídky směřuje k tvorbě uživatelského rozhraní pro uživatele, kteří se již nebudou muset zabývat tvorbou aplikace, ale na základě této práce upraví fotografie zachycující panorama do požadovaného tvaru a prostřednictvím webových služeb své panorama nahrají na konkrétní server poskytující virtuální prohlídky.

Seznam literatury a zdrojů

- [1] Pár střípků o SilverLight.
URL < <http://vavru.cz/ostatni/par-stripku-o-silverlight/>>
- [2] Silverlight - co je Silverlight?
URL < <http://interval.cz/clanky/silverlight-co-je-silverlight/>>
- [3] Seznámení s technologií DeepZoom.
URL < <http://www.maxiorel.cz/seznameni-s-technologie-deep-zoom>>
- [4] DeepZoom.
URL < [http://msdn.microsoft.com/en-us/library/cc645050\(v=vs.95\).aspx](http://msdn.microsoft.com/en-us/library/cc645050(v=vs.95).aspx)>
- [5] DeepZoom File Format.
URL < [http://msdn.microsoft.com/en-us/library/cc645077\(v=vs.95\).aspx](http://msdn.microsoft.com/en-us/library/cc645077(v=vs.95).aspx)>
- [6] Microsoft Silverlight.
URL < http://cs.wikipedia.org/wiki/Microsoft_Silverlight>
- [7] Microsoft Image Composite Editor.
URL < <http://research.microsoft.com/en-us/um/redmond/groups/ivm/ICE/>>
- [8] Adobe Flash Player
URL < http://cs.wikipedia.org/wiki/Adobe_Flash_Player/>
- [9] DevalVR.
URL < <http://wiki.panotools.org/DevalVR>>
- [10] Panorama player comparisons.
URL < <http://www.panoramaphotographer.com/comparisons/web/fpp-comparisons.html>>
- [11] Jak na panoramatický snímek.
URL < <http://wall.cz/jak-na-panoramaticky-snimek.a3.html>>
- [12] Recenze panoramatické hlavy Manfrotto 303Plus.
URL < <http://www.fotoradce.cz/recenze-panoramaticke-hlavy-manfrotto-303plus-clanekid320> >
- [13] Common Language Runtime.
URL < http://en.wikipedia.org/wiki/Common_Language_Runtime>

- [14] JIT.
URL <<http://cs.wikipedia.org/wiki/JIT>>
- [15] Panoramatické snímky.
URL <http://www.vlado.cz/3drevue/magazin3-98/panoramaticke_snimky.htm>
- [16] Creating a 360 Degree Panorama in Silverlight.
URL <<http://blog.blueboxes.co.uk/2009/08/31/creating-a-360-degree-panorama-in-silverlight/>>
- [17] Programovací jazyk C#
URL <<http://www.cs.vsb.cz/behalek/vyuka/pcsharp/text/index.html>>
- [18] Synchronizace komponent Image a MultiscaleImage.
URL <<http://cid-b1f240efb0dad5a0.skydrive.live.com/self.aspx/Silverlight/powerlawMaps.zip>>
- [19] Optimalizovaná událost pro přibližování DeepZoom obrázku pomocí myši.
URL<<http://samples.msdn.microsoft.com/silverlight/samplebrowser/index.htm#/?sref=DeepZoomCollection>>
- [20] Vývojové prostředí
URL<http://cs.wikipedia.org/wiki/V%C3%BDvojov%C3%A9_prost%C5%99ed%C3%AD>
- [21] Microsoft Visual Studio.
URL <http://cs.wikipedia.org/wiki/Microsoft_Visual_Studio>
- [22] Microsoft Expression Blend.
URL <http://en.wikipedia.org/wiki/Microsoft_Expression_Blend>
- [23] DeepZoom.
URL <http://en.wikipedia.org/wiki/Deep_Zoom>
- [24] Obrázek 3: Cylindrická projekce.
URL <<http://www.fotopraha.com/foto-praha/panorama-mostu.jpg>>
- [25] Obrázek 4: Ukázka překryvu fotografií.
URL <http://wall.cz/upload/img/jaknapanorama_foto.jpg>
- [26] Obrázek 5: Panoramatická horizontální hlava.
URL <<http://www.fotoradce.cz/soubory/clanky/320/manfrotto-303plus.jpg>>

- [27] Obrázek 9: Struktura DeepZoom snimku.
URL <http://blogs.msdn.com/blogfiles/jaimer/WindowsLiveWriter/Deepzoomprimerpart12_A83/PYRPSD_3.jpg>
- [28] Obrázek 13: Virtuální prohlídka.
URL <<http://www.360stupnu.cz/img/spheric.jpg>>
- [29] Obrázek 14: Architektura .NET
URL <<http://www.cs.vsb.cz/behalek/vyuka/pcsharp/text/resources/1.jpg>>
- [30] Obrázek 15: Překlad kódu do nativního kódu operačního systému.
URL <http://upload.wikimedia.org/wikipedia/commons/6/6f/CLR_diag.svg>
- [31] Obrázek 19: Časová osa načtení DeepZoom obrázku.
URL <<http://i.msdn.microsoft.com/dynimg/IC58225.png>>
- [32] Obrázek 20: Rozptýlený DeepZoom obrázek.
URL <<http://i.msdn.microsoft.com/dynimg/IC80918.png>>
- [33] Obrázek 21: DeepZoom pyramida.
URL <<http://i.msdn.microsoft.com/dynimg/IC141135.png>>
- [34] Obrázek 22: Přiblížení DeepZoom obrázku.
URL <<http://i.msdn.microsoft.com/dynimg/IC67074.png>>
- [35] Obrázek 31: Aplikace třetí strany.
URL <http://blogs.msdn.com/cfs-filesystemfile.ashx/_key/CommunityServer-Components-PostAttachments/00-08-87-97-14/mapsscreenshot.jpg>
- [36] Ľuboslav Lacko: Silverlight - Výukový průvodce tvorbou interaktivních aplikací. Květen 2010.
- [37] Andrew Troelsen: C# a .NET 2.0 profesionálně. Září 2006.
- [38] John Sharp: Microsoft Visual C# 2010 - Krok za krokem. Listopad 2010.
- [39] Jiří Havel: Jak fotografovat krajinu. 2007, první vydání.
- [40] Rob Sheppard, Bruce Dale: Škola fotografování Digitální technika - Nejnovější poznatky. 1. vydání.

Seznam obrázků

Obrázek 1: Použitý fotoaparát.....	2
Obrázek 2: Ukázka nevhodně použitého motivu.	3
Obrázek 3: Cylindrická projekce (obrázek převzat, zdroj č. 24).....	4
Obrázek 4: Ukázka překryvu fotografií (obrázek převzat, zdroj č. 25).	4
Obrázek 5: Panoramatická horizontální hlava (obrázek převzat, zdroj č. 26).....	5
Obrázek 6: Import snímků do aplikace Microsoft ICE.	6
Obrázek 7: Prostředí aplikace Microsoft ICE.	6
Obrázek 8: Výsledný snímek po zpracování v aplikaci Microsoft ICE.	7
Obrázek 9: Struktura DeepZoom snímku (obrázek převzat, zdroj č. 27).....	7
Obrázek 10: DeepZoom Composer – import snímků.	8
Obrázek 11: DeepZoom Composer – kompozice snímku.	9
Obrázek 12: DeepZoom Composer – Export snímku.	10
Obrázek 13: Virtuální prohlídka (obrázek převzat, zdroj č. 28).	11
Obrázek 14: Architektura .NET (obrázek převzat, zdroj č. 29).	13
Obrázek 15: Překlad kódu do nativního kódu operačního systému (obrázek převzat, zdroj č. 30).	14
Obrázek 16: XHTML stránka	16
Obrázek 17: Prostředí Visual Studio.....	19
Obrázek 18: Prostředí Microsoft Expression Blend.	20
Obrázek 19: Časová osa načtení DeepZoom obrázku (obrázek převzat, zdroj č. 31).....	22
Obrázek 20: Rozptýlený DeepZoom obrázek (obrázek převzat, zdroj č.32).	22
Obrázek 21: DeepZoom pyramida (obrázek převzat, zdroj č. 33).	23
Obrázek 22: Přiblížení DeepZoom obrázku (obrázek převzat, zdroj č. 34).	24
Obrázek 23: Znárodnění prohlížení DeepZoom obrázku.	25
Obrázek 24: Export obrázku jako kolekce.	25
Obrázek 25: Nekonečný obrázek krok 1.	29
Obrázek 26: Nekonečný obrázek krok 2.	29
Obrázek 27: Nekonečný obrázek krok 3.	29

Obrázek 28: Nekonečný obrázek krok 4.....	29
Obrázek 29: Problém hotspotu část 1.....	30
Obrázek 30: Problém hotspotu část 2.....	31
Obrázek 31: Aplikace třetí strany (obrázek převzat, zdroj č. 31).....	32
Obrázek 32: Navržená aplikace	34
Obrázek 33: Funkce ukázkové aplikace.....	35

Obsah přiloženého CD

- Zdrojové kódy programu.
- Zkompilovaný a spustitelný program.
- Série snímku zachycující panorama.
- Zpracované snímky do výsledných panoramat.
- Zpracované panoramatické snímky pro použití v technologii DeepZoom.